DOCUMENT RESUME

ED 112 917 IR 002 628

| | |
|---|---|
| TITLE | A Report on PLANIT: One Stage of Completion. Final Report. |
| INSTITUTION | Northwest Regional Educational Lab., Portland, Oreg. |
| SPONS AGENCY | National Science Foundation, Washington, D.C. |
| REPORT NO | NSF-TIE-EPP73-07319-A04 |
| PUB DATE | Aug 75 |
| NOTE | 101p. |
| EDRS PRICE | MF-$0.76 HC-$5.70 Plus Postage |
| DESCRIPTORS | *Computer Assisted Instruction; Computer Programs; *Instructional Systems; Participant Satisfaction; Program Descriptions; *Program Effectiveness; Program Evaluation; *Programing Languages; Surveys |
| IDENTIFIERS | *PLANIT; Programing Language for Interactive Teaching |

ABSTRACT

The narrative portion of this report describes the results of a recent survey of the users of PLANIT, a computer language designed to make computer-assisted instruction easier and more accessible to the user who is inexperienced with computers. The survey shows that with recent revisions in the system, user satisfaction has increased, but areas of user discontent still exist. Methods are suggested to make the system more effective. Included in the appendix of this document are a description of the PLANIT system, a discussion of its potential uses in the field of instruction, and an information brochure regarding its availability. (EMH)

# A Report on PLANIT:

# One Stage of Completion

## FINAL REPORT

## August 1975






Northwest Regional Educational Laboratory

PLANIT INFORMATION

Published by the Northwest Regional Educational Laboratory, a private nonprofit corporation. The work contained herein has been developed pursuant to a contract with the National Science Foundation.

Report No. NSF-TIE-EPP73-07319 A04



# A REPORT ON PLANIT: ONE STAGE OF COMPLETION

Final Report For:

The National Science Foundation

Grant No. EPP73-07319 A04

(Formerly EC-36800)

Erik McWilliams, Monitor

Total Grant: $99, 616

Beginning Date: Dec. 15, 1972

Ending Date: Sept. 30, 1975

Charles H. Frye, Project Director

The Northwest Regional Educational Laboratory

Portland, Oregon

28 August 1975

ABSTRACT

After an eleven year development history, the last six of which was spent developing the present machine transportable version, the ICU/PLANIT system has reached an identifiable stage of completion. Nothing more needs to be added or repaired to ready PLANIT for daily university-type operation. While there are hopes of future enhancements, these will be in the nature of add-on's rather than improvements to current features.

Following a Purdue University study of the PLANIT system performance, this project undertook a survey of all current PLANIT users to assess their reactions to the operation of the system. In general, those who are still running early obsolete versions are not too well satisfied. However satisfaction improves significantly with the later versions until the responses are very positive for the current one.

Some PLANIT lesson materials are currently available but the bulk of the authoring efforts are still in progress. Two lesson translation efforts could make a large number available to PLANIT users relatively soon.

Early interest in PLANIT waned. Since then, it is showing a steadily increasing trend until now PLANIT seems to be in great demand with new installations monthly.

TABLE OF CONTENTS

# LIST OF TABLES

# A REPORT ON PLANIT: ONE STAGE OF COMPLETION

## INTRODUCTION

PLANIT (Programming Language for Interactive Teaching) is an instructional system consisting of an author language and supporting computer programs for preparing, editing and presenting any subject matter suitable for individualized presentation to students within the constraints of the communication equipment that is currently available.

This report will briefly trace the eleven year history of the development of the computer system now called PLANIT, analyze the responses to some questions about user experiences, provide some observations regarding methods for correcting and avoiding improper operation of PLANIT which were found in the questionnaire responses to be serious concerns, and finally to suggest directions in which future PLANIT development efforts might go.

PLANIT has been described several times in numerous documents and publications. The exibit in Appendix A contains one such description and several more are cited in the bibliography so no further description was thought to be necessary here other than to say that it is a complete and comprehensive time-sharing system for instructional and problem-solving

applications, designed to be fully transportable to any commercial computer that meets certain basic size requirements.

In its present stage of development, PLANIT is a complete system. No other components need to be added to make it fully functional in a normal university computer center environment. Whatever errors yet remain in the code are apparently not serious enough to be noticed. There is no known error that remains unfixed. Such errors as are reported are fixed easily and quickly due to the dependability of the remainder of the code. Informal reports indicate that the system, in its current release of version number 2.7, is very stable and reliable, operating for months at a time with no loss of data or deviation from expected execution patterns. Purdue University has prepared an execellent analysis and evaluation of the PLANIT system.[1] It includes cost and consumptiveness data for the installation and operation of the system in its Version 1 release. The analysis also includes a report from several test sequences in which several execution errors were discovered. These have all been fixed in later releases. The report also contains an installation manual and other articles that would be of keen interest to those who might be investigating the merits of PLANIT.

The PLANIT effort has been described in several professional publications, including Creative Computing,

Datamation, Educational Technology and Impact. It has been presented in many professional meetings, including: the Association for Educational Data Systems (AEDS), the American Educational Research Association (AERA), the Conference on Computers for the Undergraduate Curricula (CCUC), the Interuniversity Communications Council, Inc. (EDUCOM), the National Institute of Chemical Engineers (NIChE), and the Western Business Education Association (WBEA). A conference was held in July, 1973 at Purdue University solely for a workshop in PLANIT, attracting 48 registered attendees. PLANIT has also been presented on numerous college and university campuses and to several military groups. More than 400 educational institutions, all major computer hardware vendors and various branches of the United States Army, Navy and Air Force have received copies of PLANIT user documentation. PLANIT information has gone to ten or more countries and the PLANIT system is operating in at least five of them.

Although PLANIT is being presented as a completed system, this does not mean that all development has ceased. On the contrary, the final section outlines several new development plans. Being complete means that it is fully usable in its present form. Not only are all those features operating correctly that were planned from the outset but several new ones were added as experience found them to be useful. Therefore PLANIT is now complete and future development will intend to enhance that which

is already operational.

PLANIT is the first known programming language and time-sharing system in existence that is fully transportable and guaranteed to be compatible. It is relatively inexpensive to install and easy to use, allowing users with no previous programming experience to do productive work after a brief orientation period. It is both a significant development in programming technology and a valuable tool for appropriate educational communities. Complete descriptions of these aspects of PLANIT can be found in publications which are cited in the bibliography.

## DEVELOPMENT AND FUNDING HISTORY

Several institutions have made sizeable investments in the PLANIT system by now. It is very difficult to give due credit to each who have contributed. Many of the investments were made with the objective to install the system for local use. While their purpose was not to contribute to the development of PLANIT, nearly all have done so by identifying problems and program errors which were not before known. The value of this information cannot be overstated.

Another grouping of contributors to PLANIT consists of those who acquired a copy of the system at a point in time and invested their own money to continue the development effort in the direction of their choosing. This includes investments from such institutions as Control Data Corporation, Michigan State University and System Development Corporation. Many of these efforts will not be reported in this section since they are tangential to the current development status of the system, i.e. the results of their work did not become a part of the system as it is today.

Forerunner of PLANIT. While the author was a graduate student at Michigan State University, he spent the summer of 1964 working at the System Development Corporation (SDC) in Santa Monica, California under a Summer Intern program.

It was during that time that he began a computer-assisted instruction project from which PLANIT eventually evolved. He programmed some statistical laboratory-type exercises in the JOVIAL language on the SDC time-sharing system. Upon returning to Michigan State, that school put up about $4,000 so that the work could be completed. SDC donated the computer time and some assistance from a Mr. Samuel Feingold, a defense systems programmer, and Mr. Joseph Rosenbaum, a researcher. With their help, a package of 25 statistical inference exercises were completed (representing about 20 contact hours of instruction), and the course was tested on several graduate students from the University of California at Los Angeles. That project was this author's dissertation study.[2]

From this experience, Mr. Rosenbaum received funding from the National Science Foundation (NSF) in the amount of $175,000 to continue the investigation of preparing similar kinds of computer scenarios for the teaching of computer programming. The author completed his work at Michigan State and joined SDC to work on the project. It was decided that a language could be devised in which to write and execute the scenarios much more efficiently than to continue using JOVIAL. Several SDC personnel contributed to the design of the language and it was coded by Mr. Feingold and the author, executing lessons early in 1966. Mr. Rosenbaum suggested the name, PLANIT.

In 1968, following the completion of the earlier project, SDC received funding from NSF in the amount of approximately $450,000 to redesign PLANIT and recode it into machine transportable form. The author became the director of that project, having as many as nine professionals on the team. The objectives for that project have been clearly stated in other reports but in summary they were:

1. To develop a running PLANIT which could be tailored by a university to meet its own needs.
2. To keep installation costs under $20,000.
3. To make PLANIT run under time-sharing or batch.
4. To make a twenty-user PLANIT run with acceptable response time in 256,000 bytes of core.
5. To make PLANIT run in 128,000 bytes of core.
6. To use ASA FORTRAN IV to achieve portability.

A PLANIT version which met these goals was demonstrated at SDC in 1970 on an IBM 360/40 (batch) computer. However, early attempts to use the system at other sites were plagued with problems due to programming errors in the system compounded by a coding complexity for transport purposes which made it nearly impossible for anyone other than the original team to modify the system or fix errors. Thus it soon became clear that PLANIT was not a system whose development could continue on any university campus.

In 1972, after installing PLANIT, the University of Freiburg in West Germany entered into a visiting professor relationship with the author for the purpose of continuing

the development of PLANIT into a reliable system and training several of their people to use it. They were running PLANIT on an RCA-type Siemens computer in a dedicated batch mode. Approximately $8,000 was spent to reach this goal and the author joined the Northwest Regional Educational Laboratory later that year to continue the development of PLANIT under a $99,000 contract from NSF.

In the new contract with NSF, only the first of the six goals was changed; this time a PLANIT was to be delivered which would be a "production" model in that following installation, it would run virtually untouched at the new site. It is no longer assumed that the receiving site will modify the system. Documentation which was designed for the earlier model to allow this has not been updated. Hence the system is expected to perform well without the continual need for maintenance. This report comes at the conclusion of the $99,000 NSF contract and at the time of this writing, PLANIT is operating with the desired reliability at sites where current versions have been installed. In addition, the NSF-sponsored Purdue analysis of PLANIT showed the performance goals to be met (with the exception of the errors which were not fixed in their version), and the installation costs were reported to be only a fraction of the earlier estimate. Their installation costs were little more than $1,300, not including the special test package that was added.

Therefore, today's completed version of PLANIT was developed for a total cost of approximately $557,000 since coding was first begun for this system. If measured from the inception of the work from which PLANIT evolved, the total investment is approximately $740,000. In either case, NSF supplied most of the necessary funding.

The interest in the current PLANIT has been much more positive than before. A PLANIT User's Group was orgainzed about three years ago and built up a mailing list numbering nearly 300 who were receiving free publications regarding PLANIT use. Since the Newsletter has gone to a subscription basis, more than 50 have elected to continue.

In order to assess the current level of interest in the PLANIT system, a questionnaire was mailed to all on the User's Group mailing list plus all in the author's correspondence file. A sample questionnaire is shown in Appendix B. The analysis of the questionnaire responses is the best available indication of the user's appraisal of the system. The analysis will be presented in a later section.

Although the future of PLANIT is still somewhat uncertain, the United States Army Research Institute has expressed their willingness to share in the future development costs. These relate to proposed additions to the system which would further enhance its capabilities. They will be described to some extent in the final section.

## ANALYSIS OF THE QUESTIONNAIRE RESPONSES

The questionnaire was designed with convenience for the respondent as its primary objective in the hope of increasihg the likelihood that it would be completed and returned.

A total of 304 questionnaires were mailed. Nine were returned undelivered. Of the remaining 295, 136 were completed and returned, for a return rate of 46%. Actually, the return rate was slightly higher due to the fact that a few of the respondents consolidated their opinions as the cover letter invited them to do. Although higher return rates are always to be preferred, the return rate for this questionnaire was quite good, especially in light of the fact that at least one questionnaire was returned for every known PLANIT installation except one, and that information was obtained by telephone (with the questionnaire still promised). It is also true that several PLANIT installations were discovered among the responses which were not expected, leading to the conclusion that there might be yet others which have not been reported.

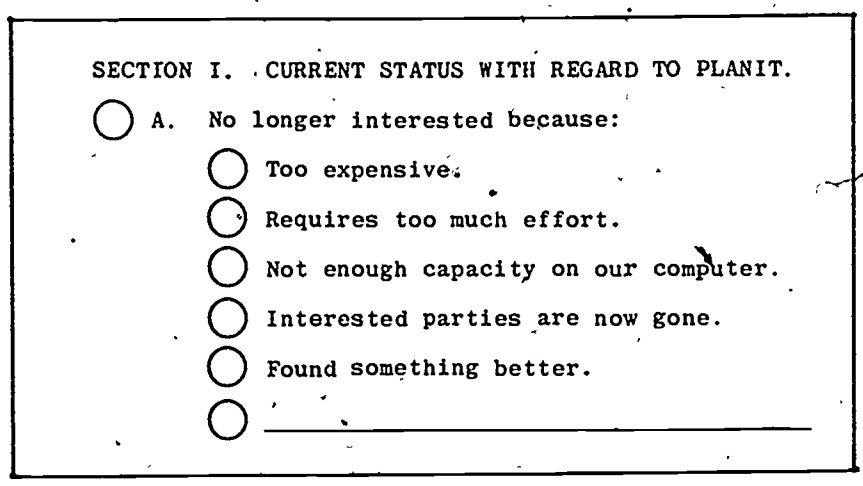
The questionnaire omitted some information that would have been useful such as the identification of the hardware, operating system, and PLANIT installation parameters. These kinds of questions were omitted due to the fear that the respondent might not have that information at hand and

would possibly lay the questionnaire aside until it could be obtained, increasing the chances that it would not be returned at all.

Because of the nature of the questions which were asked, it seems most reasonable to present and discuss the data for each question in the same order as the arrangement on the questionnaire. Therefore, each question will be reproduced in an enclosing box figure with the data and discussion immediately following. Because of the wide range of responses and response omissions, it is normally impossible to sum the tallies within categories to any meaningful number. For example, the question, "Which program time-shares your PLANIT terminals?" evoked 10 tallies for PLANIT and 21 for the host operating system. Ten plus 21 falls short of the 45 installations due to omissions, and doesn't even total 31 installations due to the fact that two respondents checked both categories. Thus, arithmetic will be used very little in the analysis of the data. Percentage conversions will be used to clarify the presentation where appropriate.

The questionnaire was organized into three sections, (I) "current status with regard to PLANIT," (II) "installation experiences," and (III) "instructional materials for PLANIT." These section titles appear in the boxes with the questions which immediately follow them.

Several of the questions (or statements) provided circles to be checked as they were felt to be appropriate. These checked circles were tallied and the counts appear just to the left or below the box, adjacent to the corresponding circle.

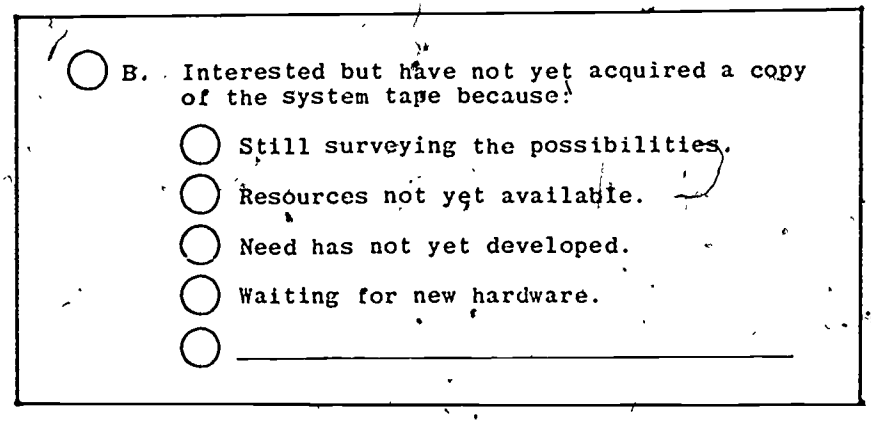



Most of these data are self-explanatory. Many of those whose names appeared on the mailing list because the contact was made mostly out of curiousity returned checks in these circles. Some even added comments to the end of the form to this effect.

There was another obvious group of respondents here who are connected with a computer-assisted instruction project on a different delivery system, who have shown commendable initiative to inform themselves of other related work. Of those who "found something better," four noted that they were using PLATO (two marking it "better," a third said, "different anyhow," and the fourth was also interested in PLANIT). APL and COURSEWRITER were named as alternatives

(although the "better" circle was not checked). Other named systems were locally developed. Operators of mini-computers probably account for most of the responses to the third sub-category. On the last line, left blank, several wrote that they had changed employers, a few said they had no need for CAI and two said they were opposed to the whole idea, one of these attaching a reprint of an article he authored to support his opinion.

It can only be guessed that a large number of those who did not return the questionnaire would probably fall into this general category.




Several who checked this category of circles also checked one or more of the category above. Judging from those who checked only this category and also from the comments that were added to the end of the form, about 36 seem to be genuinely interested in pursuing some kind of contact with PLANIT. Several wrote in dates by which machinery would be available that would allow them to try PLANIT.

55 ○ C. We have acquired a system tape of PLANIT. (Note: The remainder of this questionnaire is relevant only if you have acquired a copy of the PLANIT system tape.)

Some of the above 55 tallies were duplications in the sense that more than one response was received for some of the installations. Some of these have acquired the tape but as yet have made no effort to install PLANIT. Therefore, the number 55 is not an accurate count of installed PLANIT systems. A total of 45 installed systems could be account for.

SECTION II. INSTALLATION EXPERIENCES.

A. Version no. of the last PLANIT tape that you acquired?_____

Only 38 could report the version number of their PLANIT tape, and some of these could only report it in terms of the circumstances under which they had received it. Tapes distributed by the PLANIT Project have the version number at the head of the program listing but some of the tapes were acquired from other sources. Thus, the Version 2 tapes were reasonably clear. Prior to that were two identifiable versions with significant distribution, 1) the version which was returned from the University of Freiburg in September, 1972, was mounted by Michigan State

University and distributed by Control Data Corporation to its customers (hereafter called the CDC release), and 2) the improved Version 1, 20 copies of which were distributed at a 1973 Purdue PLANIT conference with a few more being distributed at a later time until Version 2 was ready. Thus, the version number categories of the responses (as best as they could be identified) together with the number of tallies for each are as follows:

| CDC | V 1 | V 2.1 | V 2.2 | V 2.4 | V 2.5 | V 2.6 |
|---|---|---|---|---|---|---|
| 18 | 7 | 2 | 2 | 3 | 7 | 5 |

These figures can be compared to the number of each installation that is known to exist (correcting for duplications and omissions) as follows:

| CDC | V 1 | V 2.1 | V 2.2 | V 2.4 | V 2.5 | V 2.6 |
|---|---|---|---|---|---|---|
| 16 | 7 | 2 | 2 | 3 | 6 | 9 |

These data show the pattern that has been quite obvious to some of us who are working with PLANIT, namely that the large initial circulation of PLANIT systems was due to the fact that the target sites received completely installed systems at no cost (an offer difficult to refuse), an additional flurry of activity after the Purdue conference and the formation of the PLANIT Users Group (just prior to the Purdue conference), then a sudden drop in activity but building steadily over a two-year period from 1973 to the present. At least five installations were completed

during the first five months of 1975. At least four more are in progress. Thus, the current picture is a changing one.

Accounting for the above pattern of growth is largely subjective. However, certain things seem to stand out.

The 16 CDC versions involved little if any cost or commitment. If all the data were available, the actual number of CDC copies is probably several times this number.

The next category, Version 1, is probably explainable largely in terms of novelty. For the first time, a working PLANIT system was known to exist and a mechanism had been established for obtaining it. Actually, these seven are in addition to the 20 tapes which were distributed for a small charge at the Purdue conference. It has not so far been confirmed that any of those 20 actually completed their installation. At least two were known to have started but didn't complete. The seven who did complete their installation acquired their tape elsewhere, several through the NSF-supported PLANIT Project.

The drop beginning with Version 2.1 probably shows that novelty wears off. No PLANIT lesson materials were generally available. Bugs were still being found fairly frequently. Tight budgets hit nearly all the large universities forcing reappraisals of new software ventures.

Finally, a steady growth pattern is evident. Two factors probably influence this growth. First, word has gotten out that a viable PLANIT system is indeed running at several sites. Achieving true portability in the PLANIT effort

had associated with it a high risk of failure but the word has apparently been passed among colleagues that it didn't fail. Almost every computer center director seems to know about PLANIT and his source of information seems to be another colleague. Conferences, including EDUCOM and AEDS have taken active interest in having the work presented. Magazines have carried articles and news releases. The United States Army Research Institute published a study during this time in which they screened all of the current CAI systems and chose PLANIT and proceeded to mount it on several of their computers. By this time, the system has become reliable to the extent that those concerns largely vanish. Therefore, PLANIT has increasingly become a logical choice for many who are looking for the kind of service it provides.

A second factor contributing to PLANIT's growth has certainly been the recent contracting activity by the Department of Defense for PLANIT lesson material. The RFP which was released in February, 1975 for training materials written in PLANIT caused many in both the commercial and academic fields to reassess their need for PLANIT. The bidding position of those who already had PLANIT was much better. Military installation also began serious inquiry because the training materials are going to become available. Any additional open-bid contracting for PLANIT lesson material is certain to have its affect on the proliferation of PLANIT installations. Or if NSF decides

to fund unsolicited proposals for lessons developed in PLANIT, a similar effect could be expected. Whether this proliferation of systems will generate a need for a large increase in PLANIT lesson building activity is still anybody's guess but that is certainly one possibility since the systems will exist and will need to be programmed.

Therefore, the expected growth curve for new PLANIT installations is very difficult to predict. The present 45 installations reach several hundred users but does not nearly saturate the potential market. Probably questions like "How will the system pay its way?" and "Who will maintain it?" have been asked many times and, if the answers seem promising, PLANIT will still have a bright future.

19 ◯ B. The installation effort was a success.
___ About how many man/weeks were required?

The count of the checks in the circle shown above is apparently relatively meaningless since several left that circle blank who went on to indicate quite good satisfaction with the operation of their PLANIT installation. There were probably some who did not consider their effort a success even though the system was operating.

There were even fewer responses to the number of man/weeks required. Some wrote a question mark (?) which was probably an accurate commentary for several installers,

namely that they did not keep track of the time involved. However, the data of the sixteen who did reply were very interesting. They were as follows:

1, 3, 3, 3, 4, 4, 6, 8, 8, 9, 10, 12, 16, 30, 30, 50

The range is striking. Installation of PLANIT took as little as a week and as long as almost a year. The median response was eight. The response of 12 was explained as a total of two installation efforts (two versions). Thus, only four of the 16 needed more than 10 weeks. The last two responses, 30 and 50, were for two installations abroad. Some of the above were in contact with the developer of PLANIT to clarify interface requirements. Of those who did not make that contact, the fastest installation time was three weeks, the next was four weeks. Therefore it is reasonable to conclude that good installation documentation can reduce the average installation time to a range of three to six weeks with no outside help and two weeks or less with help.

Which program time-shares your PLANIT terminals?

○ PLANIT ○ Host operating system

10 21

More than two-thirds of the respondents to this item indicated that PLANIT was being run as an object program within a host time-sharing system. Many of these were running the CDC version and had no other choice but it

does reveal the possible cause for some of the dissatisfaction that was expressed especially since those who expressed dissatisfaction were nearly all in the host time-sharing category. PLANIT is known to operate more efficiently when it does its own time-sharing. However, it is very reasonable to mount the first trial version under a host time-sharing system to simplify installation while gaining familiarity with the PLANIT interface requirements. Some have used this method as a stepping stone to an installation where PLANIT does its own time-sharing. Others implemented a fully time-shared PLANIT on the first try. The deciding factor between these two choices seems to be related to the degree of familiarity of each installer with his own operating system. If he is not sure how to implement PLANIT's interface requirements, it is usually easier under a local time-sharing system.

There are other cases where the installer must run PLANIT under a host time-sharing system, either because of computer center policy or because all remote equipment is dedicated to that system. In these cases, the interface of PLANIT to the disk files is probably the most difficult problem if the installation is to be considered successful. Unfortunately very few time-sharing systems provide the necessary services for the kinds of activities that PLANIT provides. Therefore, certain compromises must be made. Because of the number of difficulties involved, this problem will be addressed in more detail later on in this report.

Finally, some seemed to feel that PLANIT was designed to operate only under a time-sharing system and that there was little hope for improvement. It is hoped that this report will provide new understanding.

Have you updated your PLANIT installation with a later version?

| ◯ Yes | ◯ No | ◯ Not yet |
|---|---|---|
| 12 | 17 | 4 |

The responses to this item are very revealing, especially when compared to the general satisfaction with the system. The most dissatisfaction was with the older versions. Yet 21 have indicated that they have not updated. That number even falls short since only nine are running Version 2.6 leaving 36 known to be running older versions. Version 2 has been available for two years yet reports are still appearing which document errors in Version 1, errors which have long since been corrected. More than one-third of the known PLANIT installations are running the first (and least liked) version. The first step toward an improved PLANIT system performance for most sites is to update their system. Information on the available PLANIT update materials shown in Appendix C.

Have you changed the PLANIT overlay configuration from that which was on the distributed tape to better fit your needs?

○ Yes ○ No ○ Not yet

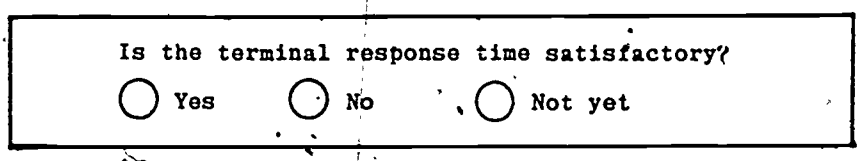
15 7 6

This question was one of those that provided insight into the potential causes of poor system performance. PLANIT was distributed until recently with only one sample overlay configuration. On a scale of 12 where Level 1 represents a high performance configuration with a relatively small number of overlays and Level 12 represents a small core, slow, heavily overlaid version, the distributed sample was at Level 9. The questionnaire responses made it clear that many have been using the sample without change despite installation instructions which recommend changes. As a result, the PLANIT tape is now being distributed with two sample overlay configurations, one for Level 3 and the other for Level 9.

Even the 15 who indicated that they did change the system were probably not all aware of the full meaning of the question. At least four of these who so indicated received their copy from CDC and in fact probably did not have the opportunity to make such changes since that step was done before the distribution.

It was significant, when comparing this question with the one asking about their general satisfaction, that

satisfaction was five times greater among those who had made changes to the overlay configuration than among those who had not. Despite all the redundancy it would cause, it may be better to send samples of all 12 configurations instead of only two in order to further encourage the implementation of the best one. On the other hand, the two current samples may give the installer more confidence in making his own adaptations since he will have the two to compare. When only one was being distributed, it was not nearly so obvious just how much work the changes would entail. There have been instances where installers have requested advice to improve their system performance and, after making changes to their overlay configuration, have been amazed at the improvement. The various factors which enter into choosing the proper overlay configuration are discussed in a later section which deals with suggestions for improving performance.




16 13 2

Opinions were about equally divided regarding their satisfaction with the speed of the terminal response. However, these data become much more revealing when compared to the version identification of their system as shown in Table 1.

| | CDC | Ver. 1 | Ver. 2 | Ver. 2.6 |
|---|---|---|---|---|
| Satisfied | 33% | 60% | 71% | 88% |
| Dissatis-fied | 67% | 40% | 29% | 12% |

Table 1. Opinions regarding terminal response time for several releases, including the one distributed by CDC; all of the Version 1 releases, all of the Version 2 releases, and the most recent version (2.6) which has been installed.

The trend of increasing satisfaction with response times is obvious. Although there have been minor improvements made to the system that would affect response times, there has not been anything significant enough to explain the above trend. Rather, it is probably due to more experience with installations and, in particular, a better fit of the overlay configuration and other variables to the target hardware. Perhaps the more recent installers have gained their experience by watching the mistakes of others. In any case, the current rate of satisfaction is much more acceptable especially when acceptable response times are maintained while overlaying from relatively slow disk packs.

Is the core usage reasonable?

○ Yes ○ No ____ Approx. how much core?

12 8

The responses to this item were mostly value judgments. It is difficult to say how much is too much. Twenty-one gave some figure indicating how much core they were using. The figures ranged from 72,000 bytes to 230,000 bytes. Most of the differences are due to the choice of overlay structure; the remainder is due to individual parameterization and compiler differences. Some suggestions for reducing core requirements are discussed later but the reduction will usually be at the expense of slower response times. Reducing core requirements generally means more and smaller overlays, which, in turn, means more delay unless the user swapped files can be moved to a faster swapping device.

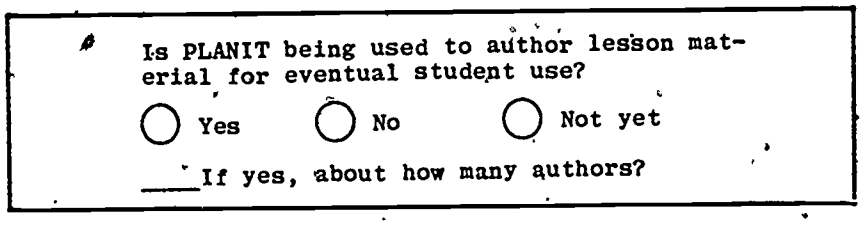
Is the disk usage reasonable?

◯ Yes ◯ No ____ Approx. how much disk?,

17 6

The amount of disk required seems not to be too much of a concern. Only seven indicated how much disk they were using and that ranged from 140,000 bytes to 8,000,000 bytes (although one reported in tracks and another in record blocks so these may or may not have been within that range). Since the consumption of disk is largely under the direct control of the installer, there is little reason to be dissatisfied with it unless the dissatisfaction is with the rate at which the lesson material consumes disk. Since PLANIT is an interpreter, the materials are stored in the

same form as it was input with as little extra control information as possible. Also, since the PLANIT language is reasonably compact, the available space is used quite efficiently. However, it is possible to install PLANIT inefficiently so that it makes working copies of the files in a host time-sharing system environment, and thus multiplies the amount of disk that would otherwise be required. Suggestions for preventing this situation appear later.




19 7 7

These responses indicate that there are 19 installations where serious lesson building activity is underway, with seven more considering it. This seems to hold the prospect for a reasonably large selection of PLANIT lesson materials to become available in the not too distant future and these will apparently be distributed free or at cost (as noted in a later item). In the blank asking how many authors, two-thirds were in the one-to-three category with the balance as high as twenty-five. In all, 118 authors were reported to be at work on PLANIT lessons. This figure would probably compare favorably with that for any other comparable system.

Are students being taught via PLANIT?

◯ Yes ◯ No ◯ Not yet

____If yes, about how many students?

____About how many hours/week?

12 7 14

Although 12 installations reported that students were being taught on the PLANIT system, only eight told how many, and only four of those told how many hours per day. The number of students being taught ranged from 12 to 450 (the University of Freiburg being the highest) for a total reported student population of about 625. The number of hours-per-day reported were 3, 3, 5 and 6. What seems most apparent is that most installations are not yet ready with their lesson materials and that a much larger student impact can be expected a little later. The fact that the largest response fell in the "not yet" category also seems to bear this out.

It has long been recognized that one of the greatest handicaps of computer-assisted instruction in general and PLANIT in particular is the lack of suitable extant lesson materials. These last two questions have produced some evidence that gains will soon be realized in this area for PLANIT.

Is PLANIT available to some user community on your computer?

◯ Yes ◯ No ◯ Not yet

____If yes, about how many terminals?

18 9 8

It is very difficult to assess the responses to this question beyond the obvious, that PLANIT is available to a user community on at least 18 computer systems. The number of terminals per system ranged from one to 180, with a median of 10. Some of the larger ones were 40, 100, 130, 148 and 180. However, these numbers could easily represent technical limitations where the system would saturate long before that number was reached. Hindsight suggests that the question should have asked how many simultaneous PLANIT terminals had been tried.

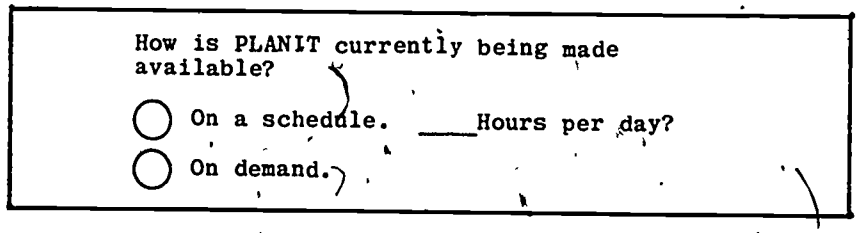
Is PLANIT being used for its calculation capability?

◯ Yes ◯ No ◯ Not yet

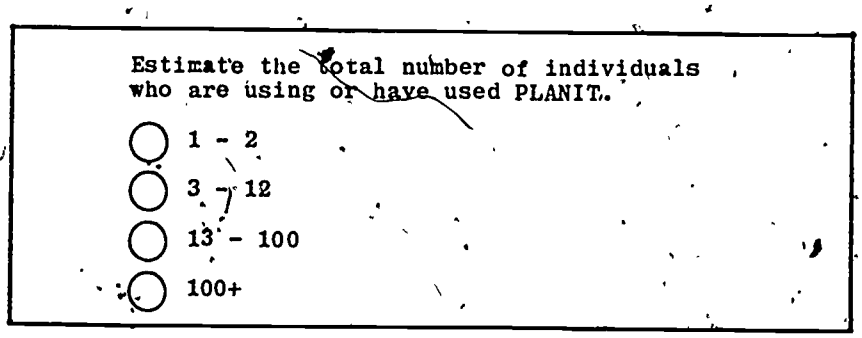
12 11 6

The true significance of the above question may not be apparent on the surface. Because of the structure of the PLANIT language, it is likely that the incorporation of calculation problems into lesson material suggests a more advanced sophistication in the use of the PLANIT language. It might give some idea of the number of serious users of the system although authors of non-

numerical lesson materials can also be serious users. Also, since integrated calculation is not generally available in other CAI systems, this gives some indication of how many perceive the need for this added capability.




Generally, where PLANIT is being operated under time-sharing, it is available on demand. Otherwise, it is scheduled. Four respondents gave the number of scheduled hours per day (even though only three checked the corresponding circle). The four were: 1, 13, 15 and 20. The reader can draw his own conclusion.




The above data are largely self-explanatory. They tend to corroborate the earlier evidence that the users of PLANIT number in the hundreds. This level of use suggests an investment well beyond the initial installation expense.

Estimate the number of PLANIT courses (or parts) that are now underway or complete.

| | |
|---|---|
| 15 | ◯ 1 - 3 |
| 3 | ◯ 4 - 12 |
| 7 | ◯ More than 12 |

These data show an encouraging amount of authoring activity. When the data are compared to the list of lessons which are now available (on a later question), the implication is that most of these courses are still in preparation. There are probably in excess of 100 PLANIT courses in progress according to the above numbers.

Are you generally satisfied with the performance of your PLANIT system?

| ◯ Yes | ◯ No | ◯ Undecided |
|---|---|---|
| 9 | 17 | 11 |

The apparent meaning of the above data suggests that PLANIT is not faring very well. The large number of dissatisfied users is certainly a matter of concern. It turns out that most of these are using the early version that CDC distributed. Table 2 shows an analysis of these data which is similar to that in Table 1 regarding user satisfaction with the response time. It is interesting to note that the pattern in the two tables is quite similar. Therefore, one of the obvious suggestions that will be

| | CDC | Ver. 1 | Ver. 2 | Ver. 2.6 |
|---|---|---|---|---|
| Yes | 7% | 20% | 57% | 88% |
| No | 79% | 20% | 14% | 0% |
| Undecided | 14% | 60% | 29% | 12% |

Table 2. Opinions regarding general satisfaction with the performance of several releases, including the one distributed by CDC, all of the Version 1 releases, all of the Version 2 releases, and the most recent version (2.6) release.

dealt with later is that users of the earlier releases should update their system. The dissatisfied users must fall into one of four categories:

1. They don't know that a better performing system is available
2. They are not using their system enough to care much about its poor performance
3. They no longer have the technical assistance available to them to accomplish the update
4. They have not yet gotten around to it

At a very minimum, it is hoped that this document will stimulate several PLANIT users to update their system. The trend of the data in Table 2 is very clear.

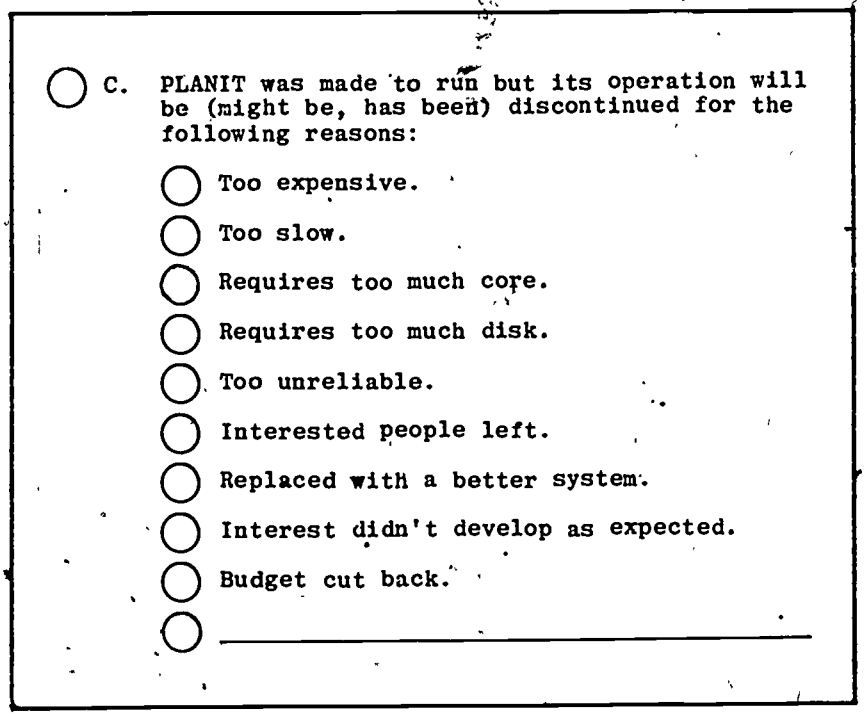
Do you intend to continue the operation of PLANIT?

| ◯ Yes | ◯ No | ◯ Undecided |
|---|---|---|
| 17 | 4 | 13 |

The trend of the responses to the above question was similar to that for the previous question. Along with the added satisfaction with the later release came less of a tendency to discontinue operation. Actually, only four had already decided to discontinue PLANIT operation, and of those four, two instances were due to shifting the installation to a different computer within the same institution. Thus, in only two cases would the PLANIT system no longer be available. However the large "undecided" vote clearly reflects dissatisfaction and further analysis shows that most of that dissatisfaction is with the earlier releases. Those who checked the "yes" circle to the above question increased from 29% for the CDC version to 88% for Version 2.6, those who checked "no" decreased from 21% for the CDC version to 0% for Version 2.6, and those who checked "undecided" decreased from 50% for the CDC version to 12% for Version 2.6. It is not too surprising that the more satisfied ones are also more apt to continue operations.

It is again easy to see the need for updating obsolete versions of PLANIT. If the latest version is giving significantly better satisfaction, then those who are contemplating

abandoning the system because their copy performs poorly should at least have the opportunity to try the later version to see if that might remedy their problem and perhaps alter their decision. There have already been some unfortunate cases where the PLANIT system has been blamed for faulty performance when in fact the particular object of the criticism had been fixed for two years or more. Thus, if poor performance is the deciding factor in the question of whether to continue PLANIT operations, then that decision is fair only if it is made on the basis of the latest release.



○ C. PLANIT was made to run but its operation will be (might be, has been) discontinued for the following reasons:

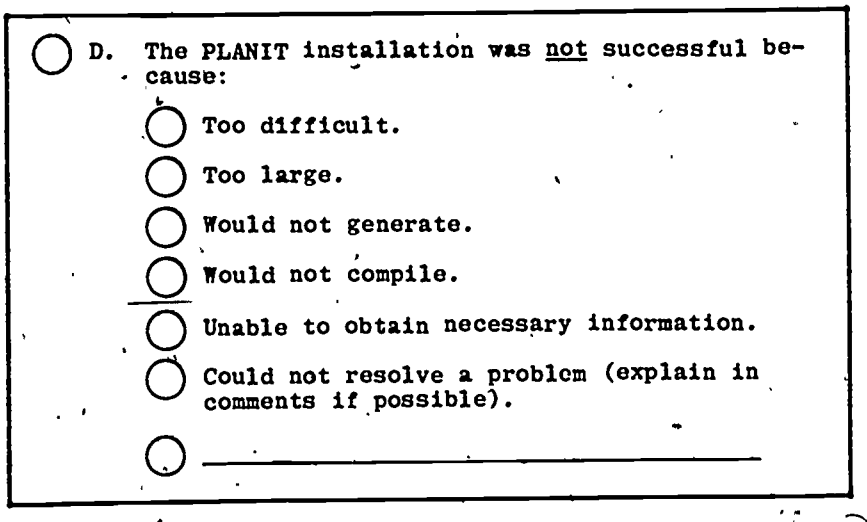
| | |
|---|---|
| 7 | ○ Too expensive. |
| 10 | ○ Too slow. |
| 10 | ○ Requires too much core. |
| 5 | ○ Requires too much disk. |
| 6 | ○ Too unreliable. |
| 4 | ○ Interested people left. |
| 6 | ○ Replaced with a better system. |
| 8 | ○ Interest didn't develop as expected. |
| 3 | ○ Budget cut back. |
| 5 | ○ _____ |

The above responses are largely self explanatory. The instructions attempted to limit the response on this item to those who had made PLANIT run but this was not

always followed accurately. Some who had not mounted PLANIT nevertheless checked such items as: Too expensive, Too slow, Too unreliable, etc. One wonders what they might have used to form the judgment. A perusal of the manuals? Knowledge of another installation?

The negative comments tended to be checked most often by those who had already expressed dissatisfaction earlier, who were also generally operating obsolete versions of PLANIT.

A few wrote brief comments in the space provided adjacent to the last circle. Most cited their concern over unavailable features or remaining bugs. One saw PLANIT only as a preliminary tool to building a superior language. Another was unhappy because he was under the impression that PLANIT would only run on CDC equipment.



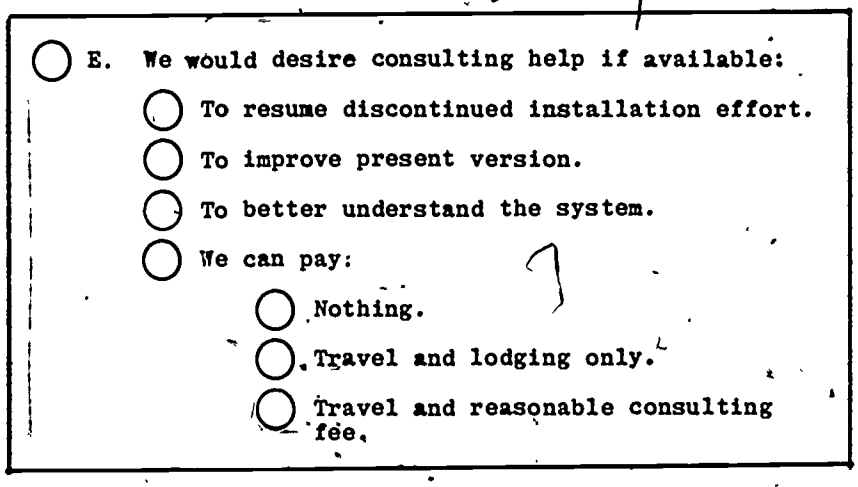
D. The PLANIT installation was not successful because:

| | |
|---|---|
| 2 | Too difficult. |
| 2 | Too large. |
| 4 | Would not generate. |
| 3 | Would not compile. |
| 2 | Unable to obtain necessary information. |
| 4 | Could not resolve a problem (explain in comments if possible). |
| 11 | _____ |

Once again the response tallies are largely self-explanatory. It is interesting to note that there has

been relatively little trouble experienced in generating the system with the PLANIT Generator program which comes as a part of the package. Considering the fact that the Generator program normally produces more than 24,000 lines of FORTRAN code which is expected to compile without errors on a variety of different computers under different FORTRAN compilers, it has performed remarkably well.

The typical comments which were added on the last line (adjacent to the last circle) were of three kinds: 1) did not have enough free time to complete the work, 2) didn't know how to handle an interface problem, and 3) waiting for appropriate hardware to arrive.




There were several who felt the need for some consulting help to improve or resume local installation efforts. However, few could pay anything toward the service. Therefore if any help is to be forthcoming, the funds will need to come from other sources.

SECTION III. INSTRUCTIONAL MATERIALS FOR PLANIT

○ A. We have one or more PLANIT lessons which may be of interest to other installations.

Subject Matter Special Entry Skills

_____

_____

_____

The library of available PLANIT lesson material is still quite small but is showing signs of great promise. Those that were reported on the questionnaire are included below. Authorship was not requested or reported so will not be included here. In a few instances the same materials were reported twice. Such is the case for the System Development Corporation's and the Army Research Institute's responses since SDC authored the materials under contract to ARI. In this case, the ARI address will be shown. The information from the questionnaires follows:

| Topics | Address of Respondent |
|---|---|
| Electrical Engineering<br>Opthamology<br>PLANIT Authoring | Gary Cagle<br>Indiana University, WCC<br>Memorial Hall 008<br>Bloomington, Indiana 47401 |
| GED Math<br>PLANIT Authoring | John Larson<br>U. S. Army Research Institute<br>1300 Wilson Blvd.<br>Arlington, Virginia 22209 |
| Medical Terminology<br>Introduction to FORTRAN<br>Introduction to JCL | Dr. Mark Leiblum<br>Katholieke Universiteit<br>Universitair Rekencentrum<br>Dreihuizerweg 200<br>Nijmegen, Netherlands |

| Topics (cont.) | Address of Respondents (cont.) |
|---|---|
| Elementary Statistics | Prof. Albert Romano<br>Computer Center<br>5402 College Avenue<br>California State Univ.<br>San Diego, California |
| Descriptive Statistics | Roger Wiley<br>Otterbein College<br>Mathematics Department<br>Westerville, Ohio 43081 |

Not listed are additional materials in Dutch from Dr. Leiblum in Nijmegen and Dr. Bert Camstra at the University of Amsterdam, also extensive material in German from the University of Freiburg. In addition, at Purdue University Dr. Franz Frederick is developing translators which will convert to PLANIT several lessons which were written in the COURSEWRITER, CLICK and PICCLES languages, and at the University of Oklahoma Winston Lindsay is also writing a COURSEWRITER III to PLANIT translator program. When these translator efforts are complete, several new lessons should become available in PLANIT.

Winston Lindsay has also obtained copies of most of the above listed lessons, using a recent issue of the PLANIT Newsletter to invite contacts from interested parties.[3]

There is obviously no such thing as a final solution to the very serious problem of the lack of available PLANIT lesson materials but the present activity of scores of authors plus the translation efforts will soon expand the above library and help to alleviate the problem.

| | |
|---|---|
| | ○ B. We are willing to make our lessons available to others on the following bases: |
| 13 | ○ Trade |
| 0 | ○ Sell |
| 10 | ○ Cost reimbursement |
| 16 | ○ Free |

The data from the above item show that the PLANIT authors are very generous in their willingness to share their work.

| | |
|---|---|
| | ○ C. We are interested in acquiring PLANIT lessons on the following bases: |
| 12 | ○ Trade |
| 4 | ○ Buy |
| 12 | ○ Cost reimbursement |
| 28 | ○ Free |

Finally, there was strong interest expressed in whatever PLANIT materials might exist, even if a purchase was required.

Several respondents made use of the "comments" space at the end of the form. Many asked questions, each of which were answered by a personal letter, though some were rather late due to the large volume. A few used the space to explain why they were not interested in PLANIT and/or computer-assisted instruction. Some described problems which they had encountered in their installation attempt. Several of these problems are specifically dealt with in the next section.

## SUGGESTED IMPROVEMENTS FOR PLANIT INSTALLATIONS

Not all of those who have installed PLANIT have been completely happy with its performance. There are several reasons which make that attitude understandable. PLANIT is a very flexible system but its suitability is focused on interactive dialogue. When used for this task, PLANIT has its greatest potential. Free dialogue lesson scenarios can be prepared in PLANIT about as easily as any other system available, requiring as few or fewer keystrokes, allowing a wide variety of grammer and numerics, providing an exceptionally good calculation capability for authors and students alike, while requiring only minimal author orientation to get started. However, it is also possible to use PLANIT on tasks for which it is poorly suited. The calculation capability, for example, while well-suited to relatively small computation tasks especially as it would relate to teaching, is not very good for large, many stepped, number crunching procedures. Using PLANIT to solve analysis of variance problems with modest sample sizes is entirely appropriate, especially in an instructional setting but to use it for large samples involving real data purely to derive the statistic would be possible though not advisable. There is little assistance which can be offered to those who use PLANIT inappropriately and are then dissatisfied.

There are several ways, on the other hand, where PLANIT can be (and has been) installed inappropriately or, at least, not as well as it might have been. Some of these situations might be helped dramatically by a few simple suggestions. This will not answer all the questions or solve all the problems but the questionnaire did reveal several installation deficiencies that can be helped.

It is probably unfortunate that the PLANIT system is as robust as it is in that it can be installed inefficiently and still operate correctly. Its image would probably be improved if it would just fail to operate at all unless it was installed as it should be. PLANIT is so completely modularized that the possibilities for choosing overlay configurations are almost limitless and will allow the system to operate whether the overlays make logical sense or not. If "thrashing" between overlays occurs, the user is usually oblivious to it, only noting that PLANIT's response time is poor. Again, unless special measures are taken, installations of PLANIT on existing time-sharing systems often cause duplicity of PLANIT's disk files and normally frustrate the reentrancy of PLANIT's coding, yet the user is only aware of the unreasonable amount of disk space that is consumed and the generally poor performance characteristics of the system. It is fair to characterize the normal user perception of the system to the effect that if the PLANIT commands seem to result in effects which are described by the manuals, then the installation is proper

were not brought to bear to assure efficient operation. Thus, as already noted, users of this system are almost universally dissatisfied with their response times. Disk consumption is abnormally high. Aborted sessions result in losing the work from the entire session even though properly installed PLANIT systems have recovery provisions which are designed to avoid this. PLANIT users on more than a third of the installations derive their impression of the system from this version. At least seven versions of PLANIT have been distributed since that one and a newer one has just become available. Any updates of systems beginning with Version 2.0 or later require no change to the interface programs, and the changes required to make Version 1 interfaces conform to Version 2 specifications are relatively minor. In general, the design of PLANIT is such that once the system has been installed, it can be updated to a later version with no recoding whatever. This is true since Version 2.0. For those who received their version from CDC and did not do their own installation work, a vastly superior Version 2.6 is operating at the Command and General Staff College in Fort Leavenworth under the 6500 SCOPE/INTERCOM system. Thus with only nine of the 45-plus PLANIT installations having the latest system release, it is not too surprising to find some dissatisfaction. You will recall that the level of general satisfaction with all aspects of the system was dramatically higher among users of the current version than among users of the initial one, and seemed

and any perceived deficiencies are due to PLANIT's design. This criticism is not intended to mean that PLANIT solves everyone's problems if only it is installed correctly. However, if its performance is poor in those tasks for which it is designed and recommended in the manuals, then it is appropriate to raise questions about the installation since it is possible, easy in fact, to install PLANIT improperly and still make it operate correctly, and also since it is being operated quite satisfactorily on a wide variety of computer equipment elsewhere. Thus, the following suggestions are being made, both to those who have already installed PLANIT and would like to improve its performance and to those who will be installing it in the hope of avoiding some of these installation hazards the results of which become so frustrating to the users.

1. Update your present system. This is one of those things that seems so obvious that it shouldn't need to be said. However unfortunate it may seem, the data show that the largest number of PLANIT systems of a given version number in operation today is also the oldest version. This version was distributed by Control Data Corporation three or more years ago, having been installed first at Michigan State University by programmers who were largely dependent on PLANIT's installation manuals for information. The use of the manuals for this purpose is of course entirely proper. However, due to the fact that this system got such wide distribution, it is unfortunate that maximum resources

to increase a little with each new release. Also, in addition to improved performance, the later release provides expanded capabilities. Each new release added something. The newest release provides expanded matrix manipulation with a compact notation not previously available. Yet all PLANIT lesson material is strictly upward compatible. It is not necessarily downward compatible though. If one is to take advantage of some of the recently developed lessons which are available, then the version numbers of the originating and receiving systems would become important since the lesson may use capabilities which were not operable on earlier systems. The first way to assure optimal performance is to be sure you are using the latest released version.

2. Adapt your overlay configuration to your own hardware. The second most apparent reason why users seem to be dissatisfied with the performance of their PLANIT system appears to stem from a poorly configured overlay structure. The data show that relatively few have changed the overlay configuration from that which was distributed on the tape. Until recently the PLANIT tape contained only one sample overlay structure and that was for a heavily overlaid, small core version. The questionnaire data indicate that most installers mounted that version without change. Installations have been found where additional core has been available for the same cost but PLANIT is limping along, shuttling overlays in unnecessarily, seriously degrading

the performance, because the installer made no change to the overlay configuration. Such changes require no more than an hour or two prior to system generation. They are very essential. This is one of two major areas where PLANIT can be installed improperly but still be made to operate correctly.

The listing documents twelve suggested overlay configurations to choose from, depending on a variety of factors including the amount of core available, the speed of the swapping medium, and the methods available for combining generated subroutines into overlay structures on the target machine after the code is compiled. The IBM 360 and 370 Link Editor lets the user build his overlay structure from any arbitrary cluster of FORTRAN subroutines. Thus, the PLANIT installer gains flexibility by breaking his PLANIT system into many small subroutines and collecting them together at linkedit time. This permits him to effect dramatic changes in the size and performance of the system without recompiling by running the linkedit step again. The CDC 6500 Segmented Loader seems to offer many of the same advantages although fewer of the systems personnel at the sites seem to be familiar with it. Without this capability, the installer must choose the overlay structure prior to PLANIT generation which will best utilize all the core that is available to him and minimize the reading of overlays from disk. Of course, faster overlaying also would make a difference. If overlays are being read fro

extended core, much heavier overlaying could be tolerated.

Having observed that so few installers were making changes to the overlay structure, the PLANIT tapes are now being sent with a choice of two overlay configurations--a so-called large core version and a small core version. Still, these are only two of the twelve recommended configurations, and a great many more than twelve are possible. The twelve which are recommended seem to make sense in terms of the amount of core that might be available, in order to maximize the probability of keeping related PLANIT modules together and minimize the potential of overlay thrashing. By giving no attention to this, poor system performance is almost inevitable.

3. Let PLANIT do its own time-sharing if possible. Of those who responded to the question about which program time-shares your terminals, PLANIT or your host operating system, two-thirds marked the latter. There seem to be many people who do not yet realize or are not convinced that PLANIT is capable of doing its own time-sharing. Yet several are running it that way even though they represent only about one-third of the installations. PLANIT is the only programming system, to my knowledge, that is a complete time-sharing system which is capable of being run as an object program within another time-sharing system. In order to accomplish this, the PLANIT system is generated for only one user and the host operating system multiplies the users.

There are several parameter options which are designed especially for those who must or prefer to run PLANIT as an object program under a host time-sharing system. These will be discussed below. In spite of these provisions, users have been discovered who have installed PLANIT under a time-sharing system but have left most or all of the parameter settings for PLANIT to do its own time-sharing. At one site, a multi-user PLANIT system was generated and was being used as an object program under a time-sharing system. There was no way for PLANIT to address more than one terminal. Yet, in all these cases, the system ran anyway. There is little wonder that efficiency is lost under these conditions. The wonder is that the PLANIT system would run at all.

To be sure, there is an overhead in PLANIT that is additive to the overhead of the host time-sharing system. Proper parameter settings will reduce this somewhat but does not eliminate it. PLANIT does its own cataloguing, controls access to files, buffers terminal data, controls access to the system, saves restart files, and a host of other things which will certainly overlap in part or in total with corresponding features in the host system. Since these things are such an integral part of the system logic, there is no practical way to make them optional. Thus, some extra overhead expense is unavoidable when running PLANIT under a host time-sharing system.

On the other hand, if PLANIT must be run under a time-sharing system (or the user prefers it), there are several options which are available to help reduce this duplication of overhead functions. These options are all available beginning with version 2.6 and some on earlier versions.

| | |
|---|---|
| CLCKTYPE=0 | This disables the quantum enforcement for the user time slice |
| NUMCH=1 | This generates a PLANIT system with one user and eliminates much multi-user code in the process. |
| FORMS=0 | This disables the user accounting in PLANIT. |
| DEBUG=0 | This eliminates code associated with PLANIT's interactive debugger the features of which are already available on many time-sharing systems. |

Another major problem encountered by those running PLANIT under a host time-sharing system is the amount of disk space that can be consumed. In order to keep core requirements down, disk files are used in real time. This means that only segments of the total file pass through core for purposes of display, update, execution for students, etc. PLANIT has complete file protection so that any file change which may be made by one user is immediately available to all other users. Authors may be making modifications to lessons while students are taking them with no conflict. PLANIT will not permit two users to be making modifications to the same lesson at the same time. However, all this is dependent on the fact that all users work from a common control table--which is not the case when PLANIT is being run under a time-sharing system. Most time-sharing systems

have been designed with great pains to insure no possibility of two users working from a common block of core. To read a disk file each time a control point is to be checked would be completely unworkable. In-slice and out-slice processing could be made to provide common core data but this is a rarity on time-sharing systems (from the user's vantage point) and in at least one case where it is available (CDC's 6500) it is extremely difficult to use.

Without the benefit of these common control points, the time-sharing user must face the problem of several PLANIT users sharing files indiscriminately. Since PLANIT can no longer referee their use, the installer must find other ways of avoiding the potential confusion.

The easiest way out is to do nothing special. In this way, PLANIT can be used by only one person at a time. Adding users will create a moderately high risk of losing lesson and record data with no chance of recovery except for the day's backup. Thus, all of PLANIT's disk library of lesson material is tied to only one user at a time, creating an intolerable storage cost and leading several respondents to conclude that PLANIT requires an "unreasonable" amount of disk space.

Another solution, one chosen by the MSU installers, is to make temporary working copies of each of the relevant files for each of the users at the beginning of his work session and, if his session completes normally, copy the temporary files back in place of the permanent ones. This avoids some of the problems but increases, rather than

reduces the disk space requirements.

A few installations have caused the large lesson files to be shared among multi-simultaneous users but have duplicated record files which are specific to the user. This is a more satisfactory solution but requires more work at the time of installation. The newest version of PLANIT will allow all users to work simultaneously from a common set of files but will add several disk accesses while the file structure is being initially defined.

The problem is in the requirement that many users have simultaneous open channels to the same disk files over long periods of time. This requirement is common for computer assisted instruction, much more so than for other typical time-sharing applications. The PLANIT time-sharing executive provides for this situation but if PLANIT is not doing the time-sharing, the checking is made inoperative, and system programmers have not found a way around this particular problem yet due to the independence among users that is built-in on the existing time-sharing systems. A fully acceptable solution would be for the time-sharing system to use an in-slice and out-slice processing feature to allow several users to share a common block of core, but this would require a change in the design of most current time-sharing systems.

Another related problem in time-sharing installations is that the reentrant feature of the PLANIT code is usually

not used, creating more disk action than would otherwise be necessary and causing a corresponding slowdown in the system performance. The only way to effectively beat this problem, if it is a problem, is to dedicate a part of core exclusively to PLANIT, and if that is done, then it is only a small step from there to let PLANIT do its own time-sharing.

A few have mounted PLANIT under a time-sharing system first to get acquainted with its operation while spending minimal time with the interface. Since most time-sharing systems contain library routines which greatly simplify the coding of PLANIT's MIOP interface subroutine, this can be a great help. A Purdue University report recommends this approach where possible. However, it is important not to stop there if at all possible, but to go ahead with the full implementation of the time-sharing version of PLANIT. This assures the most efficient operation and utilization of space. But some will not have this option and will be forced to run PLANIT under a host time-sharing system. The problems and necessary compromises have been stated. In the meantime, start making requests of the time-sharing software venders that they provide a capability for sharing core data among selected users, perhaps by in-slice and out-slice processing. PLANIT might not be the only program that could profit by it.

4. Suggestions for reducing core requirements. Some savings in core can be made by insuring that table size parameters are not larger than necessary. These include such parameters as NUMCH, INBUFF, OUTBUF, DSKSZE, VARENT, etc. They will be noted in the listing by the comment which tells how much additional core is consumed by each unit increase. These savings will be generally small but important since excess in these values results in pure waste.

Significant reduction in core consumption can be achieved by configuring PLANIT into many small overlays. The cost of this will be a decrease in performance unless the overlays can be moved very quickly. Thus, if a high-speed drum or bulk core is available for the overlays, satisfactory performance might be maintained.

Another alternative is to keep two or more PLANIT systems available, using the smaller when conditions allow. Both systems will operate interchangeably with the same set of disk files if no change is made to data parameters. A likely arrangement is to have one system with CALC routines in core for numeric-oriented lesson material and another with CALC routines in overlays for use when CALC capabilities are not so important.

5. Suggestions for reducing disk requirements. The foregoing discussion about the use of disk with time-sharing systems would relate to this but will not be repeated here.

If possible, make your MIOP request allocation of disk

only when needed rather than all at once. This does not prevent the installer from specifying a maximum that PLANIT will request.

Keep the disk space a little scarce. Some of the respondents who said that PLANIT consumed too much disk space apparently didn't realize that the amount was under the installers' control. If disk space is plentiful, there is no incentive for the PLANIT user to use the tape backup facility which is designed to free the disk of inactive material. This presupposes that the installer has implemented the tape interface for his PLANIT installation which turns out not to be true in too many cases. Thus, if disk space is to be scarce, be sure the tape interface is working properly. Small, less expensive tapes will usually be adequate.

Train the authors to be brief with their comments. Most agree that students tire more quickly of long verbage coming to the terminal particularly if the print rate is slow enough to be irritating. Much of the verbage (that is really necessary for the instruction) can be better presented on a printed sheet beside the terminal. It would save the cost of storing it on disk and may even be more pleasing to the student.

6. Suggestions for reducing costs. All of the above points have significant impact on the cost per terminal hour of running the PLANIT system. Therefore this section will just add a few suggestions which were not covered above.

Methods for cost-cutting can be related to whether PLANIT is doing its own time-sharing or is being time-shared by a host system. First, these are some suggestions for the case in which PLANIT is in control of the time-sharing.

Keep the system as busy as practical while it is running. The disk and core space allocations for PLANIT are usually fixed costs which are distributed among the users by some charging algorithm. With these costs distributed to a larger number of users, the cost per terminal/hour comes down.

Execute PLANIT directly out of lower cost extended core. This is not possible on all systems but can result in significant savings when it is. PLANIT execution is nearly always disk bound and core execution speed is seldom much of a factor since total execution times are normally quite small.

Be sure the SYSTEM WAIT call is working properly in your MIOP interface routine so that PLANIT is not charging during the times that it has nothing to do.

Use the interrupting clock (CLCKTYPE=2) over the read-only clock (CLCKTYPE=1) where you have a choice to avoid a continual flood of calls to read the clock for time-slice enforcement.

For the case where PLANIT is operating under a host time-sharing system, compare the cost of disk activity with the cost of core space. Some systems always charge for maximum core and programs which use less do not get a

price break. In that case, by all means, generate PLANIT to consume all of the available core. It is also sometimes less expensive to consume core than to overlay often from disk. This depends on the local charging algorithm.

A few installers have been troubled by the number of times PLANIT asks for the time of day. It uses the time reading to enforce the quantum in time slicing, keep user accounts, keep student performance records, and to be available to the user command. The parameter setting, CLCKTYPE=1, produces a flood of these calls. CLCKTYPE=2 produces far less. Under a time-sharing system, CLCKTYPE=0 is the appropriate setting. The number of time requests will decrease a little over CLCKTYPE=2. However, if the combination, CLCKTYPE=0 and FORMS=0 is prescribed, a dramatic reduction in time requests will result. This disables both the time slice enforcement and the user accounting. Only the timing of student response intervals will remain.

Finally, for any PLANIT installation method--in fact, for any application of any computer language--teach the PLANIT users to code their materials efficiently. There are always choices to be made in how a particular task is to be implemented. Some choices are far more efficient than others. Inefficient choices will result in increased costs per terminal/hour. It is obviously not possible to give exhaustive guidelines on efficient PLANIT coding in this document. However, it is generally more efficient to

do as much work as possible in the smallest unit of the language. For example, it is usually better to include more work in one frame than to spread it out over several where there is a choice. It is usually better to group several commands on one line instead of one-per-line. It is better to do as much in one command as possible rather than several. For example, an indexed branch command will execute more quickly than a series of IF ... branch statements which accomplish the same thing. In CALC, if a function that is to be used repeatedly results in a constant value, it is better to store it in an item defined as a constant and use that instead. A good example is found in a function which computes the mean of a series of numbers where the computed mean is used repeatedly for other computations. If the function name is used repeatedly, the mean will be re-computed each time. Where PLANIT makes special differentiated provision for authors (such as the SET function in CALC), it is usually more efficient to make use of it over the more general counterpart. Use RECYCLE sparingly. It has a lot of work to do. The latest version rearranges the lesson into optimum sequence when the UNLOCK command is given. If the lesson has been edited extensively since the last UNLOCK command was issued, give the command again. Repeating it will not hurt. If the lesson needs to be resequenced, it will be done at that time. If execution times of specific command forms in a lesson become important, perform an experiment by saving TIME before and after the

execution of each command in question to determine which is to be preferred. Branching to line labels in a Programming frame is much more efficient than branching to frame labels where the choice exists. When lessons are chained together, each should be filled to near capacity for optimal use of disk and more efficient execution. In simulation and gaming where numerical manipulations are involved, the efficiency of the algorithm can have a pronounced effect on the terminal/hour cost and on the response time of the terminal.

PLANIT installations have reported costs per terminal/hour which range from under $2.00 to more than $20.00. It doesn't seem reasonable that the same program could be responsible for such a divergent range. Yet, some charging algorithms are known to be particularly severe on certain operations (such as overlaying from disk) and when that becomes apparent, choosing a different installation option can change the cost by an order of magnitude. In at least one case, PLANIT's ability to adjust to charging irregularities caused one center to change its charging algorithm because the installers made PLANIT capitalize on a real bargain. Thus, if your costs are running abnormally high, there may be another option that would help. Other installations on similar equipment would certainly be a good resource.

## FUTURE DEVELOPMENT PLANS

Crystal balls have historically had very limited reception in attempting to view the future of CAI, not to mention PLANIT in particular. Many have told of the complete demise of CAI before now. This is not a far off look; that will be left to the soothsayers.

Rather, this section describes some fairly concrete short-term plans for developing new capabilities in the PLANIT system that may have a broad appeal. In each case the paramount goal is to maintain complete portability and compatibility among installations. It should also be noted that these new development projects are all subject to the availability of funds.

Feasibility Study for Graphics and Participation Training. The U. S. Army Research Institute (ARI) has awarded a grant for the investigation of the feasibility of adding graphics and/or a multi-terminal participation training capability to the PLANIT system. Each of these will be described separately below but is listed here to show that such a study is being initiated. The interest in actually developing the capabilities will largely depend on the outcome of this study.

Graphics. A graphics capability is being considered which would incorporate many of the display-producing

features of the common graphic devices. Thus, such features as plotting lines and circles, positioning text, cursor control, pen-up and pen-down directives, etc., are under study. Related inputs from devices such as light pens, zone detection, and analog (joy stick, mouse, etc.) input will also be considered.

There are at least three aspects of the feasibility question with regard to graphics which need further study. First, the question of portability leads to an investigation of those features which can be implemented on a wide variety of graphic display equipment, where the equipment might constitute the sole display to the user or could be an adjunct to the terminal. Second, the question of authoring directives suggests the need for careful human engineering to assure that the composing of graphics is not difficult for the author. If this problem is not properly addressed, any such new capability, no matter how versitile, would get little use and the investment would be wasted. Third, the question of need must squarely face the issue of the amount of educational advantage which is actually gained by the addition of graphic displays. So far, this need seems only to have been assumed from intuition or personal preference. A study currently being conducted at ARI has found that the empirical data showing the advantage of such displays is very sparse and are outnumbered by data which leave one in doubt about their usefulness.

Thus, the question of which, if any, of the special graphics capabilities are to be added to the PLANIT system is very much dependent on the outcome of the feasibility study. These graphic capabilities in question are in addition to the kind of keystroke drawings which can be done on any Teletype-compatible CRT display. In order to determine which new graphic capabilities are desirable, a panel of four experts will be assembled which represent four interest groups, the federal funding agency, the university, the military, and the PLANIT development project. It will be the task of this panel to come up with recommendations regarding the appropriate repertoire and formats of any new language directives for graphics. These recommendations will then be studied for technical feasibility and application to an assortment of graphic devices. The outcome will be reported and could result in a proposal to implement the recommended additions.

Multi-terminal Participation Training. Although PLANIT is already a multi-terminal system in the sense that several terminals are concurrently time-shared, it is organized in such a way that each user communicates with the lesson scenario independently of any other user. Most CAI systems use the same general format although a few of them may allow common usage of some of the numerical data. The capability which is envisioned in the feasibility study for possible implementation in the PLANIT system would permit

communication among multi-interdependent terminals such that a lesson scenario would communicate with several users at a time. This conceivably would permit such new applications as the teaching of participation, cooperative problem solving, multi-player simulations and games, etc. In comparison to the graphics study, the technology for implementing the multi-terminal capability is not nearly so difficult as the human factors problems associated with the new language directives and how they would be used by authors. Since the PLANIT system already communicates with several terminals concurrently, this capability can be passed along to authors if satisfactory language directives can be devised. This question will also be addressed by the panel which is convened for the feasibility study and the resulting recommendations may lead to future proposed efforts for the purpose of implementation.

The multi-terminal participation capability probably has its most immediate application in the training needs for large military man-machine defense systems. Since these systems already require multi-user participation, having this new capability added to PLANIT would provide the resource for this kind of training. However, it is easy to foresee applications for this capability in a variety of educational settings. The need to teach cooperative behavior has long been evident at all levels. It is reasonable to suppose that this need will not be articulated in terms of CAI system capabilities until some pilot systems

are available for experimentation. Thus a capability of this kind could add a new dimension to the direction of CAI research and development. This is a case where no new hardware technology is needed, rather the current components can be reconfigured to stimulate a new direction for investigation; that is, new for CAI research.

## CONCLUSIONS

The PLANIT system is complete in its current stage of development. Several installers using a variety of equipment have reported that installation proceeds pretty much as described and that the system runs well. It performs a full complement of tasks to provide a CAI capability for a computer facility. User satisfaction as reported on a recent questionnaire is good for the latest version, having improved consistently with each new release. Interest and installation activity is also accelerating. Even though the number of currently reported installations is only in the forties, each of these represents a community of users such that hundreds of people are getting hands-on exposure to PLANIT, and this figure is expected to mushroom when military installations begin to duplicate their installations on a large scale. In addition, letters of interest from many universities suggest that installation activity may continue to increase for a time. Therefore, PLANIT is certainly a viable CAI software package at the moment.

On the other hand, even though the PLANIT system is complete, new development efforts are to be expected as long as the level of interest is high and resources are available. In fact, new developments to the system could be the key to its maintaining its viability for a much longer period of time. It is true that CAI is still

largely in the hands of the researchers and will be put into general use only when the right combination of costs and capabilities is found. In the meantime, in order to get the best expected return on prior investments, it is important to make every reasonable effort to keep the development of the system at the forefront of the current state-of-the-art in order that there will be some attractive options for new directions in the research investigations. If this is not done, PLANIT will become obsolete before it is put to work in a practical setting. Therefore the development of PLANIT will not be finally finished until it is conceded to be obsolete.

PLANIT has an undeniable edge over comparable systems in its portability and low-cost installation. Its language features are considered by many to be superior. If its lack of graphics has been a handicap, then the addition of a graphics package that is as portable as the rest of the system should be a definite advantage. Although a few have suggested that, because of its coding, PLANIT would be too inefficient to run well, statistics have not borne this out. Several installations are reporting costs which compare favorably with any of the CAI systems currently available. However, due to variations in local charging algorithms, objective comparative data is almost impossible to obtain.

Having devoted several years to the development of PLANIT, and as the author of this report, I am well

satisfied with the level of acceptance which PLANIT currently enjoys and I see many more challenges yet to be achieved before my interest in the project diminishes.

## FOOTNOTES

1. Frederick, T. J. and F. J. Frederick, A Final Report on Analysis and Evaluation of the PLANIT System, NSF Grant No. GJ-35633, Purdue University.

2. Frye, C. H., Teaching Statistical Inference by Computer: Problem Simulation for Hypothesis Testing and Evaluation, Michigan State University, 1967. Univ. Microfilms (Order no. 67-14,503).

3. Smith, L. B. (ed.), PLANIT Newsletter, Vol. 4, No. 2, July, 1975.

# PLANIT REFERENCES


Atkinson, R. C. and Wilson, H. A. (Ed.), Computer-Assisted Instruction: A Book of Readings, Academic Press, New York, 1969.

Bennik, F. D., Hoyt, W. G. and Butler, A. K., A CAI Course on Constructing PLANIT Lessons: Final Report, TM-5364/000/00, System Development Corporation, Santa Monica, July, 1974.

Bennik, F. D. and Frye, C. H., PLANIT Author's Guide, TM-L-4422/001/01, System Development Corporation, Santa Monica, October, 1970 (Reprinted by the Northwest Regional Educational Laboratory, Portland, Oregon).

Butler, A. K. and Frye, C. H., PLANIT Language Reference Manual, TM-L-4422/002/01, System Development Corporation, Santa Monica, October, 1970 (Reprinted by the Northwest Regional Educational Laboratory, Portland, Oregon).

Computing and the Decision Makers, Proceedings of the EDUCOM 1974 Spring Conference, Princeton, New Jersey, April, 1974.

Derks, K., CALC: Interaktief Rekenen aan de Terminal, Universitair Rekencentrum, CAI Project, Nijmegen, Nederland, June, 1974.

Frederick, T. J. and Frederick, F. J., A Final Report on Analysis and Evaluation of the PLANIT System, NSF Grant No. GJ-35633, Purdue University.

Frye, C. H., "CAI Languages: Capabilities and Applications," Datamation, (14,9), September, 1968.

Frye, C. H., "Choosing a Computer Language," in Post, D. L. (Ed.), The Use of Computers in Secondary School Mathematics, Entelek, Inc., Newburyport, Mass., 1970.

Frye, C. H., "PLANIT: The Portable CAI System," Creative Computing, November-December, 1974.

Frye, C. H., "Using a Computer for Instruction," Impact, (4,2), Winter, 1969.

Haefner, K., Zwischenbericht fur das Jahr 1972 und Uebersicht ueber die Erste Phase des Projektes von 1970 bis 1972, Projekt CUU, Universitat Freiburg, 78 Freiburg, West Germany, June, 1973.

Johnson, C. D., "Implementation of PLANIT at the U. S. Army Research Institute for the Behavioral Sciences," PLANIT Newsletter, (4,2), July, 1975.

Rosenbaum, J. and Bennik, F. D., "CAI at System Development Corporation," Educational Technology, February, 1969.

Williams, T. G. and Frye, C. H., "An Instructional Application of Computer Graphics," Educational Technology, (8,11), June, 1968.

# APPENDIX A

## PLANIT DESCRIPTION

# PLANIT

## A Computer Assisted Instruction Language System

The objective of this article is to acquaint the reader with the Computer Assisted Instruction Language System known as PLANIT. The article includes a description of the system, its language components and how they work, and some of the language's special features. Then there is a discussion on why the system is machine-transferable.

The PLANIT Language System grew out of a previous project at Michigan State University in which the computer was used to teach skills in testing statistical hypotheses and making inferences. Later, System Development Corporation developed PLANIT on an early time-sharing system.

In 1968, the National Science Foundation contracted SDC to rewrite the PLANIT system so that it could be used on a variety of university and college computers and the development effort has since moved to the Northwest Regional Educational Laboratory in Portland, Oregon. Thus, the current machine-transferable version of the PLANIT system.

The acronym PLANIT stands for Programming LANguage for Interactive Teaching. It consists of four basic modes of operation: 1) lesson-building, 2) calculation, 3) execution, and 4) system. PLANIT is both interactive and interpretive. In the lesson-building mode, as soon as a frame has been completely typed in, it can be executed and checked out; in the calculation mode, as each statement is entered, the answer is calculated and returned to the user. The system mode is used for lesson and record management.

PLANIT is a frame-oriented language and has four frame types: Question (Q), Multiple Choice (M), Decision (D), and Programming (P). The frames are divided into groups consisting of lines. The Q and M frames each have four groups and the D and P frames each have two groups. (Each frame has a similar group one, its identifying frame number and the label of that frame if it has been given one.) The language was designed to take into account the fact that the user may make mistakes; although there are specified forms that are outlined in the Reference Manual, other forms of input will be accepted by the system.

Ease as Objective

The basic objective of the design of the language was to make it easy to use. This was accomplished in several ways. First, the system is built to be interactive, so that when the user is building frames, he can be continually prompted as to the correct next input. Second, the user is given a complete set of on-line editing capabilities to alleviate fear of making mistakes and to make it easier to correct parts of the lesson whose logic does not work as expected. One of the thoughts guiding the designers of PLANIT is that a lesson designer should not have to know anything about computers to be able to write CAI lessons. Previously, CAI lesson authors needed two disciplines, one in their subject matter and one in computer programming.

PLANIT was designed to bridge that gap so that only one discipline would be needed. For this reason, all PLANIT commands are in a natural language. For example, to declare a function the user begins his input with the word FUNCTION: to insert a frame into the lesson sequence or a group into a frame or a line into a group, the user types the letter I followed by the appropriate

frame, group, and line specification. During lesson building, PLANIT is continuously giving short messages as prompts to the user, so that with very little knowledge of the language, a user can start writing in, and learning about, PLANIT. (Lessons can also be prepared offline on cards.)

The user is in the command or lesson-building mode as soon as he logs into PLANIT. After accepting the log-in value, PLANIT prints ENTER COMMAND. To begin building a lesson, the user types an A for append. As he builds his frames, PLANIT continually prompts him for the next input.

There are seven editing commands available to the user if he needs to make corrections. They are A for append, D for delete, I for insert, E for edit, P for print, M for modify and S for search. Any of these may be used with any combination of frames, groups, or lines; for M and S column specifications can be made. Some examples will clarify these statements. If we input, 2, 4, 3, A, we mean append to Frame 2, group 4, line 3, the lines that are to follow. If we have built a frame and left out a group (which is perfectly legal) but have now decided to insert that group, we might input 5, 3, I which would insert Group 3 into frame five. We can print ranges of frames or several groups within a frame, or only group 3's within a range of frames, or any other combination possible. The same variety of options is available using the D (delete) command.

Change, Transfer Possibility

The M editing command changes character strings, and may also use a column specification; for example: If we said 3-5, 2-4, 1-99, 2-7, M, PLANIT would prompt us with /FROM/TO/. If our next input was /AA/B/ then all occurrences

of AA would be changed to B in the range of frames, groups, lines, and columns specified. The M command can contract or expand a line or do exact character transfers. It can also be used to delete single or multiple characters. The S command may also have column specifications. Entering this command causes PLANIT to print /MATCH/. After the user input, such as /KEYWORD/, PLANIT will print all of the frames, groups, and lines in which that character string exists. The format of the printout will be a series of frame number, group number, line number.

The user need not complete writing his entire lesson in one sitting. PLANIT allows each log-in value to have one unnamed lesson associated with it. If after a short session, the user logs out, his unfinished lesson is automatically saved; when he relogs into the system, he will automatically be associated with his old unnamed lesson. He may choose to use the PLANIT SAVE command and give his unfinished lesson a name. Then, on relogging, he must first issue the GET command and the lesson name to be associated with that lesson. At any time that he is associated with a lesson, he may enter the execution mode, even if the lesson is incomplete.

There are several other commands available to the user in lesson building mode: DISPLAY, which displays selective student records; CLEAR, which resets the student records to a clean start; RESTART, which erases all work and returns the user to the status he was in just after he logged in; and CALC, which transfers the user into the interactive Calculation mode.

The PLANIT calculation language (hereafter called Calc) is in itself a BASIC-like language. It can be used as a powerful one-line calculator or the statements can be combined into a Programming frame, which will be discussed

later. Calc has the following built-in functions: all of the trig functions, FACTorial, COMBinatorial, square root (SQRT), LOG, ALOG, RANK, SORT, INVERT, SUM, PROD, TRUNCATE, NORMAL, ABSOLUTE, and a random number generator. The user may define his own functions, matrices, or items.

All of these may be reserved for either author or student use or for use by both student and author. The ALIGN function allows PLANIT to print graphs. There are also other words, all natural English, that provide special capabilities for the user; for example, the word ROUND, followed by a number, sets the round-off format for displaying results. There is a FOR statement that can be used in repetitive calculation and a compact notation for matrix manipulation. In short, Calc is a complete algebraic interpreter. Used as a desk calculator, it is more powerful than most. For example, it is possible to write a one-line statement that will do the work of complete programming algorithms.

The Calc mode has no limits on the number of dimensions that a matrix contains--two is just as easy to handle as six or seven. Any expression that reduces to a single number may be used wherever a single number can be used; that includes matrix subscripts. A one-line statement using a FOR statement is capable of generating a complex multi-lined graph.

Response Processing

PLANIT has several response processors, which are controlled through Calc statements; the main ones are KEYWORD, PHONETIC, TEXT, FORMULAS, WITHIN and WAIT. KEYWORD has three modes of operation. When KEYWORD is in the ON condition, PLANIT will accept as a correct answer anything that matches the specified author answer, even if it is accompanied by other words;

for example, if the answer were "blue" and the student input was "it is blue" and if KEYWORD were ON, then the answer would be counted as correct. With KEYWORD in the ALL condition, the answer could be unordered; for example, if the answer were "George Washington," the ON condition would accept either "George last name Washington," or "President George Washington," but not "Washington George." The ALL condition would accept all three as correct. An author can also use KEYWORD with a number. If KEYWORD were set to the number 3, and the answer was "President George Washington," then any of the three words given, in any order, with any other words intervening, would be considered a correct response.

The use of PHONETIC causes PLANIT to encode both the author's and the student's answer phonetically before attempting a match. This allows misspelled correct answers to be judged as correct, such as George Washingtun for George Washington. Studies have indicated that PHONETIC is about 85 percent effective.

TEXT lets the author find root words or substrings within words such as matching on the word START for answers which contain words like START, STARTED, STARTING, RESTART, etc.

With FORMULAS in the ON condition, PLANIT will evaluate formulas algebraically. An example would be if the answer was the formula for the area of a triangle "1/2*B*H" and the student input was "(H*B)/2.". In a straight character match, the answer is incorrect, but with FORMULAS ON, the answer given by this student would be counted as correct.

Tolerance for Answers

WITHIN allows the author to specify a numeric answer within any tolerance he wishes. For example, if the answer specified is 2 WITHIN .5, PLANIT will accept as correct any number between 1.5 and 2.5.

WAIT followed by a number (for seconds) sets a time limit within which the student must type his answer. Otherwise the lesson will proceed accordingly. If WAIT is not used, the student has an indefinite time limit.

Groups /Tag Matching

The group two of the Q frame is where the text that is to be presented is input. During execution, the text will appear exactly as input, (with the exception of special formatting control symbols that are available to the lesson author).

The answers are specified in the group three. Each answer is associated with a tag. Alphabetic tags are used for non-numeric answers and number tags for numeric answers. The correct answer is specified by a plus sign next to the tag; there can be more than one plus tag in the group, meaning there is more than one correct answer, or there may be no plus tag, meaning that the frame is a neutral frame with no correct answer.

Any Calc statement may be used in this group if it is preceded by a zero tag. With reference to the response processors, the statement might be "O KEYWORD ON." The PLANIT special response processors may be turned on and off interchangeably at this time. As an example, one may turn a response processor on for some of the answers and not for the rest.

The group four is for actions. PLANIT has four action commands: F, C, R, and B. F stands for feedback; using this command, the author can print any text that he wishes. He may let PLANIT randomly choose either a positive or a negative feedback message, depending on whether or not a correct answer tag was matched. The C is used for any calculation statement; or alternately, the C may be used to that PLANIT will print the correct answer. In this case, it would be the first tag in the group that had a plus tag.

The R stands for repeat. It may be followed by any textual materials, just as the F, or, if it is not followed by text, PLANIT will print the predefined message WRONG TRY AGAIN. In both cases, PLANIT asks the student for another response to the same question. The B is for branching. PLANIT allows the author to branch to a frame label, a frame number, or another lesson. When branching to a frame number, the author can use as the number of the frame any Calc expression that reduces to a single number or may designate multiple branch points, one of which is selected by a variable value. Any number or combination of action commands can be executed based on the match of one tag.

All of the action commands are associated with an answer tag. More than one tag can be grouped together as a tag field, so that the same actions may occur for several different tags. If any action commands are at the beginning of the group and have no tag field before them on the line, they will be executed regardless of the tag the student matched. A Q frame sample is shown in Figure 1 and the student interaction that might result is shown in Figure 2.

```
FRAME 1.00 (Q)

G2. TEXT
WHO WAS THE THIRD PRESIDENT OF THE UNITED STATES?

G3. ANSWERS
A+ THOMAS JEFFERSON
0  PHONETIC ON
B+ THOMAS JEFFERSON
0  KEYWORD ON
C  JEFFERSON
D  WASHINGTON
E  ADAMS
F  LINCOLN

G4. ACTIONS
A F:
B F:RIGHT, BUT YOU SPELLED IT INCORRECTLY.  IT IS THOMAS JEFFERSON.
C R:JEFFERSON IS RIGHT BUT GIVE BOTH NAMES.
D R:NO, HE WAS THE FIRST.  TRY AGAIN.
E R:ADAMS WAS THE SECOND, TRY AGAIN.
F R:HE WAS THE SIXTEENTH, TRY AGAIN.
- R:NO, NOT EVEN IN THE FIRST THREE.  TRY AGAIN.
- R:
DEF- R:YOU ARE GUESSING. ONE MORE TRY.
DEF- F: C:
```

Figure 1. Sample of a PLANIT lesson frame.

```
PLEASE LOG IN***CHF
PLANIT TERMINAL NO. 1
ENTER COMMAND
*GET PRESIDENTS
IDENTIFY YOURSELF***CHF

WHO WAS THE THIRD PRESIDENT OF THE UNITED STATES?

*JOHN ADAMS
ADAMS WAS THE SECOND, TRY AGAIN.

*JEFERSON
JEFFERSON IS RIGHT BUT GIVE BOTH NAMES.

*THOMAS JEFERSON
RIGHT, BUT YOU SPELLED IT INCORRECTLY.  IT IS THOMAS JEFFERSON.
```

Figure 2. Sample of student interaction in a PLANIT lesson.

The M frame is identical to the Q frame, with the following exceptions. The answers with the tag are printed out to the students, without the plus sign. PLANIT will accept only one of the tags as an answer. If other than a tag is input, PLANIT will print the message CHOOSE ONE OF THE ABOVE LETTERS.

Student Performance Data and the D Frame

The following questions are automatically answered and the information is kept by PLANIT as the student executes the lesson: Has he seen the frame? Did he answer correctly or incorrectly? Exactly what tag did he match? Did he go into the Calc mode? If he did, which functions did he use? How long did he take to answer the question? Anywhere in the lesson the author may have D frames, which may query the student record with Boolean capabilities for any combination of the above information; he may query his own counters in combination with the above. The outcome of the D frame is usually a branch or choice of branches, depending on the outcome of the queries. All of the action commands are available to the author, with the exception of the R.

Programming Frame

The P frame is a Programming frame. In this frame, the author composes a series of Calc statements that make up a program. This frame can also be used as a subroutine. If a frame branches to a P frame and in the execution of that P frame a RETURN statement is executed, the execution of the lesson will pick up at the instruction following the branch. This could be another action command or the next frame in sequence from the calling frame. Also,

the P frame may be used as a subroutine through the Calc mode. The frame is called by typing the word "GOTO" and then either the frame number or label. The return in this case is made back into the interactive Calc mode.

Just as a P frame can be used as a subroutine within a lesson, so an entire lesson can be used as a subroutine to other lessons making it possible to build entire courses from a sequence of calls on appropriate lessons.

User Identification

PLANIT keeps two separate identifications for the user. The first is the user's log-in value. This is the identification that he inputs when he first becomes interactive with the system. When an author wishes to save a lesson he has built, another identification is asked for by the system. This is his user value. After a user logs into the system, he may request a lesson by typing GET and a lesson name. After he issues the GET command, he is asked for his user identification. There are three types of users on the system: a read/write author, a read/only author and a student. After the issuance of the GET command and the acceptance of the user identification, PLANIT decides which type of user is operating at that terminal. If the user identification is identical to the one that was used when the lesson was built, then the user is an author.

If the log-in value also matches, he is the read/write author; otherwise he is a read/only author. If the user identification does not match the one associated with the lesson, the user is defined as a student. Both types of authors (after "getting" a lesson) are put into the lesson-building mode; if the user is a student, he is immediately put into the execution mode. The execution of the lesson follows the instructions as set forth in the frames by the

author, including all branches. The student need never know when he is being branched, even if he is branched to another lesson. The student records are kept automatically.

At any time that PLANIT is awaiting an input from the student, in answer to a frame, the student may choose to go into the interactive Calc mode. The author has control over this, however, and can PROHIBIT or ALLOW the student to enter the Calc mode. Or the author can ALLOW the student the Calc mode but PROHIBIT selective functions. The student has the full capabilities of the Calc mode under normal conditions, with a few exceptions consisting of special author-only functions such as the control of the response processors. Even those, however, may be allowed to the student, so that if the author wants to test the lesson and see it as the student is going to see it, but wishes to use some functions that would usually be denied a student, he may do so.

Student Help

The student has available to him a built-in review function that allows him to review sequences of the lesson. This is under his control, and he may ask to review at any time. The only precondition is that the sequence he will see must start with an author-defined labelled frame that he has already seen. He need not have seen the entire sequence, however. The author has the ability to prohibit the use of the review function, or allow it. A similar GOTO function can permit the student to browse through the lesson at selected entry points. The author may predefine places from which the lesson will start if the student ends the session before he has actually finished the lesson. This is especially important in sections of a lesson where preparatory information is needed before a student

can proceed. If the student should quit in the middle of the sequence, then this capability could be essential to allow him to resume the sequence appropriately again.

Prompting

PLANIT gives prompts whenever it is expecting input from the user. Minimum prompting is an asterisk. If the user does not know from the message what PLANIT is asking for, then the user may type a question mark (?), getting from PLANIT a more complete message. There may be a third linked message to elaborate even further. For example, PLANIT prompts the user in the lesson-building mode by printing the message "Q/M/D/P." By typing a question mark, the user would get "(Q)uestion/(M)ultiple choice/(D)ecision/(P)rogramming." By typing another question mark, the user would get "ENTER FRAME TYPE."

Program Language Defined

The first task in building the system was to define the programming language in which to code. To do that, those features from ASA FORTRAN IV that were independent of both machine and operating systems were chosen.

Then those features that were needed to do efficient coding such as in subroutines, string manipulation, overlaying, and equivalencing were built back into the language. One of the most important added features was the ability to write variables into the code that would be converted to constants before compile-time. An item such as the dimension for an array must become

a literal constant before attempting to compile; this makes it possible to use the same variable in programmed checks on that array.

When the value is finally filled in, the program and the data will then agree. In fact, one of the major obstacles to machine independence is the difficulty of making the program agree with the rearranged data format imposed by the new machine. The PLANIT System accomplished this by specifying both the program and data formats in terms of the same set of variables. Thus, the language in which the coding was done was not really FORTRAN. A new language was devised in which to write PLANIT, a meta/language if you will, called ICU/FORTRAN. (ICU for Instructor Computer Utility.)

Installation is accomplished through a system generation process in which a generator program (also delivered with PLANIT) produces a customized ASA FORTRAN IV VERSION of PLANIT according to user-supplied parameters. Almost any medium scale (or larger) computer is adequate, having at least a 24-bit word size. The system will require from 20,000 to 40,000 words of core and several hundred thousand characters worth of space on a disk-like device. Figure 3 shows a simplified installation process.

The PLANIT system will support a variety of terminals and has all of the required time-sharing software built-in. The prior existence of time-sharing on the target computer is not necessary. However, PLANIT does operate compatibly with existing time sharing systems.

PLANIT is operational at such major institutions as Michigan State University, Washington State University, Purdue University, Northwestern University, the U. S. Command and General Staff College, and others. There are more than 50 installations in the United States and Europe on computers from most major manufacturers.

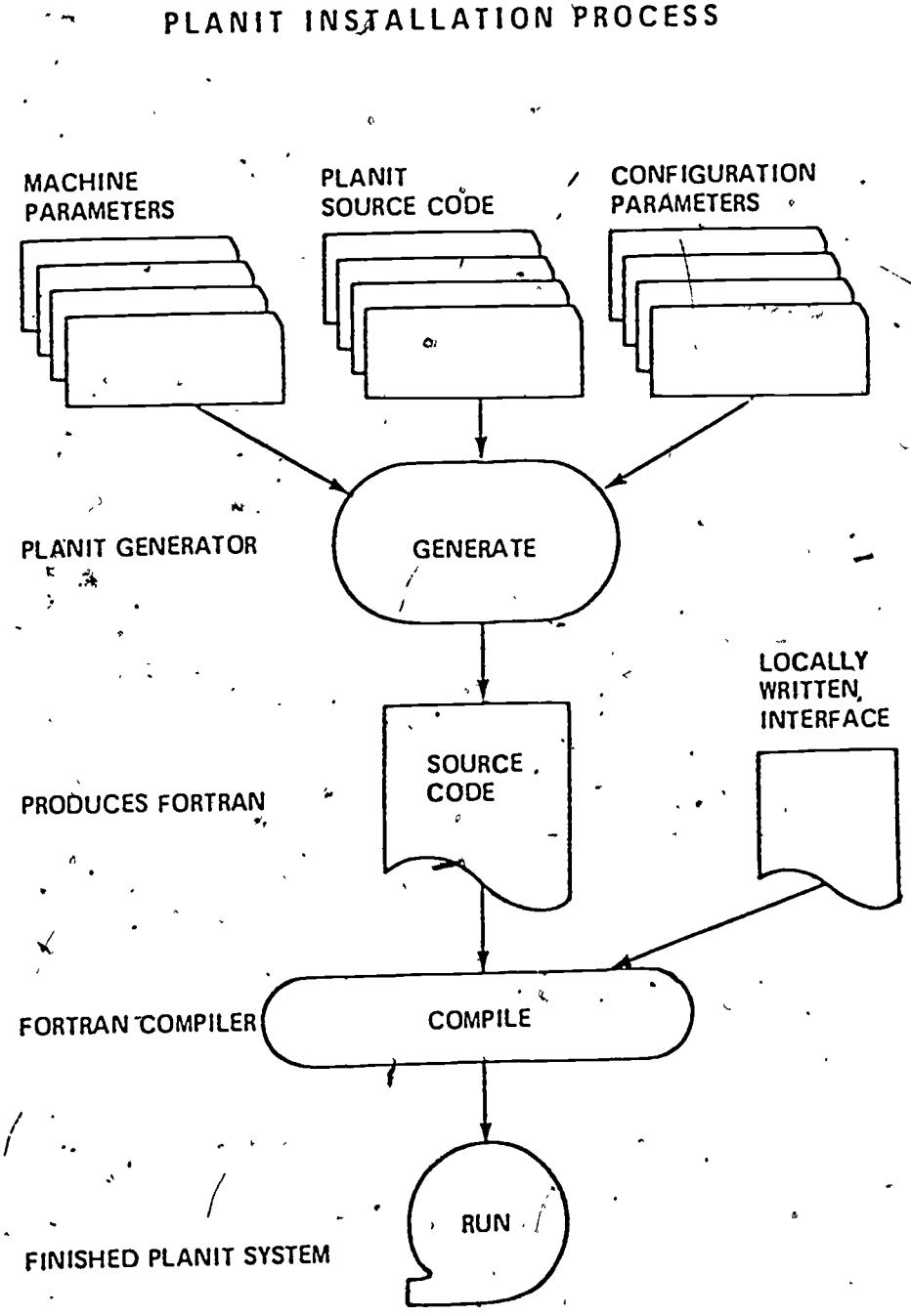



Figure 3. Steps in the PLANIT installation process.

A PLANIT users group attempts to maintain communication among interested parties and provides a vehicle for lesson exchange.

Full documentation and installation materials are available from:

Project PLANIT
Northwest Regional Educational Laboratory
Lindsay Building/710 S. W. Second Avenue
Portland, Oregon 97204

# APPENDIX B

## SAMPLE QUESTIONNAIRE

Northwest Regional Educational Laboratory
Lindsay Building · 710 S.W. Second Avenue
Portland, Oregon, 97204 · Telephone (503) 224-3650

February 12, 1975

Dear Colleagues:

I am sorry to impose another questionnaire upon you, supposing that it might be the third one this week. Let me assure you that this information is needed for my final report to the National Science Foundation on the present impact of PLANIT.

Your name was chosen because it appeared on the roster of a PLANIT conference, on the mailing list of the PLANIT Newsletter, or in my correspondence file. I realize that you might have no direct contact with PLANIT but there is a little information regarding your anticipated interest that I would appreciate anyway.

The questionnaire appears to be highly structured. This was done only for your convenience. These are not intended to be forced choices. Check whichever boxes you feel apply. If you have the time to write additional information, be assured that it will be used.

Two copies of the questionnaire are enclosed along with a mailing list by institution. If you happen to know of an institution that should have received a questionnaire, you would be doing me a real favor if you would forward the extra copy to them. If not, discard the extra one.

Others at your institution may be receiving this questionnaire. If one returned questionnaire represents several respondents, I would appreciate your noting that. However I would also value a questionnaire from each respondent and will gladly sort out the redundant information at this end.

Let me thank you in advance for your effort. A short turnaround would mean a great deal to me so that I can finish the report in reasonable time. Results will appear in the PLANIT Newsletter. If I can be of help with regard to PLANIT, please let me know.

Sincerely,

Charles Frye

Charles H. Frye

Enc.

# A QUESTIONNAIRE ON THE CURRENT IMPACT OF THE PLANIT SYSTEM

## SECTION I. CURRENT STATUS WITH REGARD TO PLANIT.

◯ A. No longer interested because:

- ◯ Too expensive.
- ◯ Requires too much effort.
- ◯ Not enough capacity on our computer.
- ◯ Interested parties are now gone.
- ◯ Found something better.
- ◯ _____

◯ B. Interested but have not yet acquired a copy of the system tape because:

- ◯ Still surveying the possibilities.
- ◯ Resources not yet available.
- ◯ Need has not yet developed.
- ◯ Waiting for new hardware.
- ◯ _____

◯ C. We have acquired a system tape of PLANIT. (Note: The remainder of this questionnaire is relevant only if you have acquired a copy of the PLANIT system tape.)

## SECTION II. INSTALLATION EXPERIENCES.

A. Version no. of the last PLANIT tape that you acquired? _____

◯ B. The installation effort was a success.

_____ About how many man/weeks were required?

Which program time-shares your PLANIT terminals?

◯ PLANIT ◯ Host operating system

Have you updated your PLANIT installation with a later version?

◯ Yes ◯ No ◯ Not yet

Have you changed the PLANIT overlay configuration from that which was on the distributed tape to better fit your needs?

◯ Yes ◯ No ◯ Not yet

Is the terminal response time satisfactory?

◯ Yes ◯ No ◯ Not yet

Is the core usage reasonable?

◯ Yes ◯ No _____ Approx. how much core?

Is the disk usage reasonable?

◯ Yes ◯ No _____ Approx. how much disk?

Is PLANIT being used to author lesson material for eventual student use?

◯ Yes ◯ No ◯ Not yet

_____ If yes, about how many authors?

Are students being taught via PLANIT?

◯ Yes ◯ No ◯ Not yet

_____ If yes, about how many students?

_____ About how many hours/week?

## SECTION II. (Cont.)

Is PLANIT available to some user community on your computer?

◯ Yes ◯ No ◯ Not yet

_____ If yes, about how many terminals?

Is PLANIT being used for its calculation capability?

◯ Yes ◯ No ◯ Not yet

How is PLANIT currently being made available?

◯ On a schedule. _____ Hours per day?

◯ On demand.

Estimate the total number of individuals who are using or have used PLANIT.

◯ 1 - 2

◯ 3 - 12

◯ 13 - 100

◯ 100+

Estimate the number of PLANIT courses (or parts) that are now underway or complete.

◯ 1 - 3

◯ 4 - 12

◯ More than 12

Are you generally satisfied with the performance of your PLANIT system?

◯ Yes ◯ No ◯ Undecided

Do you intend to continue the operation of PLANIT?

◯ Yes ◯ No ◯ Undecided

(If you have no complaints about your PLANIT system, you may skip to Section III.)

◯ C. PLANIT was made to run but its operation will be (might be, has been) discontinued for the following reasons:

- ◯ Too expensive.
- ◯ Too slow.
- ◯ Requires too much core.
- ◯ Requires too much disk.
- ◯ Too unreliable.
- ◯ Interested people left.
- ◯ Replaced with a better system.
- ◯ Interest didn't develop as expected.
- ◯ Budget cut back.
- ◯ _____

◯ D. The PLANIT installation was not successful because:

- ◯ Too difficult.
- ◯ Too large.
- ◯ Would not generate.
- ◯ Would not compile.

SECTION II. (Cont.)

- ◯ Unable to obtain necessary information.
- ◯ Could not resolve a problem (explain in comments if possible).

_____

◯ E. We would desire consulting help if available:

- ◯ To resume discontinued installation effort.
- ◯ To improve present version.
- ◯ To better understand the system.
- ◯ We can pay:
  - ◯ Nothing.
  - ◯ Travel and lodging only.
  - ◯ Travel and reasonable consulting fee.

SECTION III. INSTRUCTIONAL MATERIALS FOR PLANIT

◯ A. We have one or more PLANIT lessons which may be of interest to other installations.

Subject Matter Special Entry Skills

_____

_____

_____

◯ B. We are willing to make our lessons available to others on the following bases:

- ◯ Trade
- ◯ Sell
- ◯ Cost reimbursement
- ◯ Free

◯ C. We are interested in acquiring PLANIT lessons on the following bases:

- ◯ Trade
- ◯ Buy
- ◯ Cost reimbursement
- ◯ Free

Thank you so much. Please give your name, institution, city and zip code (if the following is blank or incorrect).

Name: _____

Institution: _____

City: _____ Zip: _____

Please return this questionnaire in the enclosed envelop or mail to:

Dr. Charles H. Frye
Project PLANIT
The Northwest Regional Educational Laboratory
710 S.W. Second Avenue
Portland, Oregon 97204

COMMENTS:

PLANIT INSTITUTIONAL MAILING LIST BY STATE AND COUNTRY

Arizona
A. State Univ.
Benson Union High Sch.
Maricopa Technical Coll.
Scottsdale Community Coll.
Univ. of A.

California
C. State Univ., Chico
C. State Univ., Northridge
C. State Univ., San Bernadino
City College of S.F.
CDC
DCA Assoc.
H-P
Honeywell
Litton Data Sys.
Stanford Univ.
SDC
UC Irvine
UCLA
UC San Diego

Colorado
C. State Univ.
Ent AFB
Peterson Field
Univ. of C.

Delaware
Univ. of D.

Florida
F. State Univ.
Univ. of South F.

Georgia
CDC
G. State Univ.
Valdosta State College
Univ. of G.

Illinois
CDC
I. Board of Higher Ed.
I. State Univ.
Northwestern
Northern I. Univ.
SRA
Univ. of I., Champaign
Univ. of I., Urbana

Indiana
CDC
I. Univ., Bloomington
I. Univ., Indianapolis
Purdue Univ.

Iowa
Drake Univ.
Univ. of I.

Kansas
Fort Leavenworth
Univ. of K.

Louisiana
Univ. of Southwestern L.

Maine
Univ. of M.

Maryland
U.S. Naval Academy
CDC

Massachusetts
DEC
Ginn & Co.
M. State College
Project LOCAL
Univ. of M.

Michigan
Central M. Univ.
Dearborn Public Schools
Eastern M. Univ.
M. State Univ.
Univ. of M.
Wayne State U.
Western Mich. Univ.

Minnesota
Bemidji State Coll.
CDC
Sperry-Univac
Univ. of M.

Mississippi
Keesler AFB

Missouri
Univ. of M., Columbia
Univ. of M., Rolla

Montana
M. State Univ.

Nebraska
Univ. of N.

New Jersey
Univ. Heights Campus

New Mexico
Eastern N.M. Univ.
Western N.M. Univ.

New York
Clarkson College of Tech.
Columbia Univ.
Cornell Univ.
FDR VA Hospital
S.U.N.Y. at Buffalo
S.U.N.Y. at Purchase
S.U.N.Y. at Stony Brook

Ohio
Battelle Columbus Labs
Bowling Green State Univ.
CDC
Otterbein College

Oklahoma
Univ. of O.

Oregon
N.W. Regional Educ. Lab.
O. State Univ.
Univ. of O.

Pennsylvania
Baldwin Senior H.S.
Dickinson College
Lehigh Univ.
Temple Univ.

Texas
Alvin Junior College
T & M Systems
L.T.V. Aerospace
Philco-Ford
Randolph AFB
Southern Methodist Univ.
Univ. of T.

Utah
Courseware, Inc.

Virginia
College of William & Mary
Honeywell
MITRE Corp.
U.S. Army Research Inst.
V. Tech. Univ.

Washington
W. State Univ.

Washington, D.C.
Dept. of Army
Dept. of Navy
George Washington Univ.
National Science Foundation

Wisconsin
Univ. of W., Eau Claire
Univ. of W., Madison

Australia
Center for the Study of H. E.
Government of Western A.
Univ. of Melbourne
Univ. of New South Wales

Canada
Concordia Univ.
Dalhousie Univ.
McMaster Univ.
Sir George Williams Univ.
Univ. of Calgary
Univ. of Lethbridge
Univ. of Windsor
Univ. of Winnipeg

Canal Zone
Division of Schools

England
Univ. of Aberdeen
Univ. of Essex

Israel
IBM

Italy
Csata-Bari Univ.
Institute of Computer Applications

Netherlands
Catholic Univ.
CDC
Free Univ.
Fysich Laboratory
Rijks Univ.
Sara-Amsterdam
Univ. of Amstordam
Utrecht State Univ.

Switzerland
Computer-Wissenschafton

West Germany
CDC
Heidelberg Rohabilitation Institute
Siemens, Inc.
Telefunken, Inc.
Univ. of Cologne
Univ. of Freiburg
Univ. of Bremen
Univ. of Tubingen

# APPENDIX C.

## PLANIT INFORMATION BROCHURE

# INFORMATION REGARDING THE AVAILABILITY OF PLANIT

Recent interest in PLANIT has generated a large number of requests for information, documents and source code. Although our documentation effort is still underway which will eventually produce correct and readable manuals, the Laboratory can now supply the needed information some of which is still in draft form. The PLANIT source code is also available on magnetic tape.

Document Requests. The Northwest Regional Educational Laboratory has reprinted by permission two of the PLANIT manuals from the SDC (System Development Corporation) set:

The PLANIT Author's Guide, 344 pages, price: $13.00

The PLANIT Language Reference Manual, 368 pages, price: $13.50

The above manuals are two of a six-volume Tech Memo set which was produced at SDC in 1970. The other four volumes are now obsolete for the current version of PLANIT.

There are several supplementary documents which will be combined into one and printed as an attachment to the Language Reference Manual. Until that is ready, Xerox copies of all these materials will be made available for an additional $3.50. This will include a copy of the Purdue PLANIT Installation Manual. Although the form is not yet as nice as we would like it to be, the information is all there.

Source Code Requests. The Laboratory is prepared to send a magnetic tape copy of all of PLANIT's master files such that the installer will have all the necessary source code to generate a PLANIT system for his own computer. The tape can be sent in 7 or 9 track with a choice of density, blocking and parity. We recommend odd parity, blocking of 40 cards per record and 800 BPI for 9 track (556 BPI for 7 track). The tape will consist entirely of card images and a chart will be enclosed which describes the tape format and character codes.

The total cost of the tape will be $35.00 if we supply the blank, or $20.00 if you supply it. In the latter case, the blank should be mailed to:

Mr. Richard Million
Computing Center
Washington State University
Pullman, Washington 99163

When mailing a blank, please enclose a return label. The desired tape parameters may be listed on the blank tape but please also include them in a letter to the Laboratory since the writing of the tape is initiated from here via a remote terminal.

Address all tape requests to this Laboratory at the address shown at the end of this form. The request should contain the desired tape parameters (density, parity, blocking) and indicate whether a blank has been sent to Mr. Million. Send all payments to this Laboratory.

Before purchasing a tape unnecessarily, please note that the word size of your computer must be 24 bits or larger in order to accomodate PLANIT (which rules out most mini computers).

If you do not supply the blank tape, the Washington State Computing Center will provide one and bill the Laboratory. The tape will be a good one but not necessarily new. The PLANIT tapes which we send are guaranteed to be free from parity errors or we will do it over again.

Installation Assistance. By now, PLANIT has been installed on a fairly large sample of computers. Some have been able to use the installation efforts of others to good advantage.

Consulting can be provided by the Laboratory at a daily rate plus expenses, making the services of the developer of PLANIT available on-site. In that case, the local site will be expected to provide someone who is thoroughly knowledgeable about the operation of that computer system.

Many installation questions can be answered over the telephone. If no additional work is involved, these can be handled without charge and installers are encouraged to make such calls after noon (Pacific time) to: Dr. Charles H. Frye, (503) 246 9960. This contact can be especially useful in the interim until the installation documents are brought up to date.

PLANIT Users' Group and Newsletter. A PLANIT Users' Group has been in existence for over two years and now publishes a newsletter quarterly, available for an annual subscription rate of $4.00 (individual - North America), $5.00 (individual - Overseas) and $10.00 (Institutional), from:

Dr. Lyle B. Smith
S.C.I.P. - Acacia
Stanford University
Stanford, California 94305

Payments. PLANIT has been developed on public money and is being distributed by the Laboratory at no profit. Every attempt is being made to keep the costs low. Therefore, the prices quoted are valid only if the payment is made in full at the time of the request. All prices include handling and postage at the economical mailing rate, and are subject to change when the cost to the Laboratory changes.

If you find it necessary to be billed, a $5.00 invoicing fee per order will be added.

Allow two to four weeks for delivery due to postal delays. For faster delivery, include first class or airmail postage The shipping weights are:

| | |
|---|---|
| Author's Guide | 48 oz. |
| Language Reference Manual | 50 oz. |
| Xerox materials | 18 oz. |
| Magnetic tape | 72 oz. |

Our goal is to make the entire PLANIT package available as conveniently and economically as possible, making a total computer software system available for an incredibly low price--less than $100--a price which has no equal for a software system of PLANIT's magnitude

PLANIT Project
The Northwest Regional Educational Laboratory

Mail inquiries to:

Dr. Charles H. Frye
Director, PLANIT Project
The Northwest Regional Educational Laboratory
710 S.W. Second Avenue
Portland, Oregon 97204

Selected PLANIT articles can be found in:

Creative Computing, Nov-Dec, 1974

Educational Technology, June, 1968

Datamation, September, 1968